

N

PAT-NO: JP411085890A

DOCUMENT-IDENTIFIER: JP 11085890 A

TITLE: FINANCIAL INSTITUTION SERVER, SECURITY SYSTEM FOR CLIENT
WEB BROWSER, AND METHOD THEREFOR

PUBN-DATE: March 30, 1999

INVENTOR-INFORMATION:

NAME
CHANG, SHEUELING
MARKS, STUART

ASSIGNEE-INFORMATION:

NAME	COUNTRY
SUN MICROSYST INC	N/A

APPL-NO: JP10150509

APPL-DATE: April 22, 1998

INT-CL (IPC): G06F019/00, G06F013/00, G06F017/60, G09C001/00, G09C001/00
, G09C001/00

ABSTRACT:

PROBLEM TO BE SOLVED: To make communication between a financial server and a user who is related to a web browser safe by including some security functions in the web browser and the web server.

SOLUTION: A financial server 102 includes an audit trail 122 which traces each transaction. An HTML form 124 performs transmission to a web browser 216 in a cipher format or without performing special formattings. The browser 216 returns a message 143, including form data to the server 102 in an enciphered format, in a format which includes the digital signature of a user and a time stamp and in a format which includes the digital signature of the user and a time stamp and is enciphered or without performing special formattings. The browser 216 is provided with an encipher procedure 220, a time stamp procedure 228, a digital signature procedure 230, and a random key generating procedure 232.

COPYRIGHT: (C)1999,JPO

(19)日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11)特許出願公開番号

特開平11-85890

(43)公開日 平成11年(1999) 3月30日

(51)Int.Cl. ⁸	識別記号	F I	
G 0 6 F 19/00		G 0 6 F 15/30	M
13/00	3 5 1	13/00	3 5 1 G
17/60		G 0 9 C 1/00	6 2 0 A
G 0 9 C 1/00	6 2 0		6 4 0 Z
	6 4 0		6 6 0 B

審査請求 未請求 請求項の数28 O L 外国語出願 (全 60 頁) 最終頁に続く

(21)出願番号 特願平10-150509

(22)出願日 平成10年(1998) 4月22日

(31)優先権主張番号 08/841430

(32)優先日 1997年4月22日

(33)優先権主張国 米国 (US)

(71)出願人 591064003

サン・マイクロシステムズ・インコーポレーテッド

SUN MICROSYSTEMS, INCORPORATED

アメリカ合衆国 94303 カリフォルニア州・パロアルト・サンアントニオロード・901

(72)発明者 シェウエリング チャン

アメリカ合衆国 カリフォルニア州

95014 クーバーティノ レグナートロード 22345

(74)代理人 弁理士 中村 稔 (外6名)

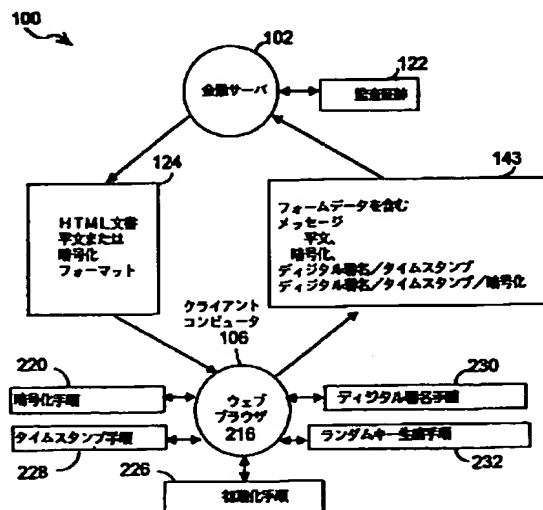
最終頁に続く

(54)【発明の名称】 金融機関サーバ及びクライアントウェブブラウザ用セキュリティシステム及び方法

(57)【要約】

【課題】 金融サーバと、ユーザとの間のコミュニケーションを安全にするシステム及び方法を提供する。

【解決手段】 金融取引処理システムは、クライアントコンピュータに関連する複数のユーザと公衆網を通して接続されている少なくとも1つの金融サーバを含む。各ユーザはウェブブラウザを通して金融サーバにアクセスする。ウェブブラウザは暗号化キーを生成し、HTMLフォームを暗号化及び解読し、HTMLフォームにデジタルサインし、タイムスタンプを添付する能力を有している。HTMLフォームを含む金融サーバ転送ウェブページは、金融取引を表す。HTMLフォームは、インカミングフォーム及び戻りフォームのフォーマットを指定する拡張を含む。HTMLフォームは、暗号化フォーマット、ユーザのデジタル署名及びタイムスタンプを含むフォーマット、及びユーザのデジタル署名及びタイムスタンプを含む暗号化フォーマットで伝送できる。金融サーバは、ユーザの会計、ユーザのデジタル署名、取引のタイムスタンプ、及び取引のテキストを含む監査証跡を通して各処理済取引を追跡する。



【特許請求の範囲】

【請求項1】 コミュニケーションリンクによって相互接続されている少なくとも1つのクライアントコンピュータと、少なくとも1つのサーバコンピュータとの間の金融取引の伝送をコンピュータで実現する方法において、上記方法は、

(a) 上記サーバコンピュータから1つまたはそれ以上のHTML文書を受信するステップ、を備え、上記文書の部分集合はフォーマット指示文を含み、上記フォーマット指示文の各第1の部分集合は戻りメッセージを上記サーバコンピュータへ伝送するために使用されるアウトゴーイング伝送フォーマットを指示し、上記フォーマット指示文の各第2の部分集合は上記HTML文書を受信するために使用されるインカミング伝送フォーマットを指示し、上記フォーマット指示文の上記第1及び第2の各部分集合は少なくとも1つの暗号化技術に関連付けられており、

上記方法は更に、

(b) 受信したHTML文書からのフォームデータを、上記戻りメッセージ内へ挿入するステップと、

(c) 上記フォームデータに1つまたはそれ以上の暗号化技術を適用するステップと、を備え、上記適用される各暗号化技術は、上記受信したHTML文書に関連付けられた関連フォーマット指示文内に識別されており、

上記方法は更に、

(d) 上記戻りメッセージを上記サーバコンピュータへ伝送するステップを備えていることを特徴とする方法。

【請求項2】 上記フォーマット指示文は、暗号化、デジタル署名及びタイムスタンプ、及びデジタル署名及びタイムスタンプを用いた暗号化からなるセットから選択される請求項1に記載の方法。

【請求項3】 上記ステップ(c)は、

(1) 第1の暗号化キーを生成するステップと、

(2) 何等かの挿入されたユーザ関連情報を含むフォームデータを、上記第1の暗号化キーを用いて暗号化するステップと、

(3) 上記第1の暗号化キーを上記戻りメッセージに添付するステップと、

(4) 上記第1の暗号化キーを、第2の暗号化キーを用いて暗号化するステップと、を更に含む請求項1に記載の方法。

【請求項4】 上記ステップ(c)の(1)は、上記第1の暗号化キーを生成するために、ランダムキーシーケンス発生器を使用するステップを更に含む請求項3に記載の方法。

【請求項5】 上記ステップ(c)の(4)は、上記サーバコンピュータから上記第2の暗号化キーを入手するステップを更に含む請求項3に記載の方法。

【請求項6】 上記ステップ(c)の(1)は、

上記生成ステップの前に、指定されたユーザに関連付けられたデジタル署名を上記戻りメッセージ内に格納するステップを更に含む請求項3に記載の方法。

【請求項7】 上記ステップ(c)は、デジタル署名及びデジタル署名検証子を生成するステップと、

上記サーバコンピュータに上記デジタル署名検証子を供給するステップとを更に含む請求項6に記載の方法。

【請求項8】 上記ステップ(c)は、上記戻りメッセージ内にタイムスタンプを格納するステップを更に含む請求項1に記載の方法。

【請求項9】 少なくとも1つのサーバコンピュータにコミュニケーションリンクを通して接続されている少なくとも1つのクライアントコンピュータを含むコンピュータシステムのデータにアクセスするためのウェブブラウザにおいて、上記ブラウザは、

複数のHTML文書を格納するメモリを備え、

上記HTML文書の部分集合は、上記HTML文書内に含まれるフォームデータを戻すために使用される暗号化プロトコル指示文を含み、

上記ブラウザは更に、

上記サーバから上記HTML文書の種々のものを検索し、且つ戻りメッセージ内に含まれる上記フォームデータ内にユーザ関連情報を挿入するブラウジングメカニズムと、

上記戻りメッセージ及び上記HTML文書を、指定された暗号化プロトコルに従って処理する暗号化処理メカニズムと、を備え、

上記ウェブブラウザは、上記ブラウジングメカニズムを使用して上記戻りメッセージを上記サーバコンピュータへ伝送し且つ該サーバコンピュータから受信し、また上記暗号化処理メカニズムを使用して上記HTML文書及び戻りメッセージに対して1つまたはそれ以上の暗号化プロトコルを実行し、上記クライアントコンピュータに関連付けられた意図したユーザが上記HTML文書を受信することを可能にし、上記サーバコンピュータへ伝送される上記戻りメッセージの安全な伝送を提供可能にすることを特徴とするウェブブラウザ。

【請求項10】 上記暗号化処理メカニズムは、上記戻りメッセージを暗号化し、上記HTML文書を解読する暗号化処理メカニズムを含む請求項9に記載のウェブブラウザ。

【請求項11】 上記暗号化処理メカニズムは、上記戻りメッセージにサインし、上記HTML文書を認証するデジタル署名処理メカニズムを含む請求項9に記載のウェブブラウザ。

【請求項12】 上記デジタル署名処理メカニズムは、タイムスタンプ処理メカニズムを含む請求項11に記載のウェブブラウザ。

【請求項13】 上記暗号化処理メカニズムは、1つま

3

たはそれ以上の暗号化キーを作成する暗号化キー生成メカニズムを含む請求項9に記載のウェブブラウザ。

【請求項14】 上記暗号化キー生成メカニズムは、1つまたはそれ以上のランダムセッションキーを作成するランダムキー生成メカニズムを含む請求項13に記載のウェブブラウザ。

【請求項15】 金融サーバと共に使用するためのコンピュータコードメカニズムを含むコンピュータ可読記憶媒体において、

上記コンピュータコードメカニズムは、金融サーバからHTML文書を検索するブラウジングメカニズムを備え、

上記HTML文書はフォームデータを含み、上記ブラウジングメカニズムは上記フォームデータ内にユーザ関連情報を挿入して上記金融サーバへ伝送される戻りメッセージに添付し、上記金融サーバから検索される上記HTML文書の部分集合は上記金融サーバへの戻りメッセージを交換するために使用される暗号化プロトコル指示文を含み、

上記コンピュータコードメカニズムは更に、上記HTML文書を指定された暗号化プロトコルに従って処理する暗号化処理メカニズムを備え、

上記コンピュータコードメカニズムは、上記ブラウジングメカニズムを使用して上記金融サーバから上記HTML文書を受信し且つ戻りメッセージを上記金融サーバへ伝送し、また上記暗号化処理メカニズムを使用して上記HTML文書に対して1つまたはそれ以上の暗号化プロトコルを実行し、意図したユーザが上記HTML文書を受信することを可能にし、上記金融サーバへ送信される上記戻りメッセージの安全な伝送を確保可能にすることを特徴とするコンピュータ可読記憶媒体。

【請求項16】 上記暗号化処理メカニズムは、上記HTML文書を解読し、上記戻りメッセージを暗号化する暗号化処理メカニズムを含む請求項15に記載の媒体。

【請求項17】 上記暗号化処理メカニズムは、上記戻りメッセージにサインし、上記HTML文書を認証するデジタル署名処理メカニズムを含む請求項15に記載の媒体。

【請求項18】 上記デジタル署名処理メカニズムは、タイムスタンプ処理メカニズムを含む請求項17に記載の媒体。

【請求項19】 上記暗号化処理メカニズムは、1つまたはそれ以上の暗号化キーを作成する暗号化キー生成メカニズムを含む請求項17に記載の媒体。

【請求項20】 上記暗号化キー生成メカニズムは、1つまたはそれ以上のランダムセッションキーを作成するランダムキー生成メカニズムを含む請求項19に記載の媒体。

【請求項21】 金融取引処理のためのコンピュータネットワークにおいて、上記ネットワークは、

4

1つまたはそれ以上のユーザに各々が関連付けられている複数のクライアントコンピュータと、

少なくとも1つの金融サーバと、を備え、

上記金融サーバは、

金融取引を表す複数のHTML文書を格納するメモリを備え、

上記各HTML文書はフォームデータを含み、上記HTML文書の部分集合は上記クライアントコンピュータと上記サーバコンピュータとの間で金融取引を交換するために使用される暗号化プロトコル指示文を含み、

上記金融サーバは更に、上記フォームデータを暗号化し、且つ各受信した上記HTML文書を解読するために使用される1つまたはそれ以上の暗号化処理メカニズムと、

上記クライアントコンピュータからのコミュニケーションを管理するサーバメカニズムと、を備え、上記コミュニケーションの部分集合は上記フォームデータを含む戻りメッセージを含み、上記サーバメカニズムは受信した上記各戻りメッセージに関連付けられた上記暗号化プロトコル指示文を翻訳し、且つ各受信された上記各戻りメッセージを1つまたはそれ以上の対応する暗号化プロトコルメカニズムに従って処理する命令を含むことを特徴とするコンピュータネットワーク。

【請求項22】 上記暗号化処理メカニズムは、上記フォームデータを暗号化し、且つ上記HTML文書を解読する暗号化処理メカニズムと、機密情報を上記金融サーバと交換するように登録された各ユーザに関連付けられたユーザ公開キー情報を含むユーザ情報データベースと、

クライアントコンピュータから受信した各デジタルサインされたコミュニケーション内のデジタル署名を、上記ユーザ情報データベース内に格納されている対応するデジタル署名がもしあれば、それに従って検証するデジタル署名検証メカニズムと、

上記クライアントコンピュータから受信したデジタルサインされたコミュニケーションのレコードを、上記受信した各コミュニケーションが、上記金融サーバと機密情報を交換するために登録されている関連ユーザによってデジタルサインされていることを検証するのに充分なだけ格納する監査証拠と、を含む請求項21に記載のネットワーク。

【請求項23】 上記暗号化プロトコル指示文は、暗号化、デジタル署名及びタイムスタンプ、及びデジタル署名及びタイムスタンプを用いた暗号化からなるセットから選択される請求項22に記載のネットワーク。

【請求項24】 上記各クライアントコンピュータは、上記金融サーバから上記HTML文書にアクセスするためのウェブブラウザを含む請求項22に記載のネットワーク。

【請求項25】 上記ウェブブラウザは、上記フォーム

10

20

30

40

50

データを暗号化し、上記アクセスされたHTML文書を
 解説する暗号化処理メカニズムを含む請求項24に記載
 のネットワーク。

【請求項26】 上記ウェブブラウザは、上記戻りメッ
 セージにサインし、上記HTML文書を認証するディ
 ジタル署名処理メカニズムを含む請求項24に記載のネッ
 トワーク。

【請求項27】 上記デジタル署名処理メカニズム
 は、タイムスタンプ処理メカニズムを更に含む請求項2
 6に記載のネットワーク。

【請求項28】 上記ウェブブラウザは、1つまたはそ
 れ以上の暗号化キーを作成する暗号化キー生成メカニ
 ズムを含む請求項24に記載のネットワーク。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】本発明は電子通信システムに
 関し、詳しく述べれば、取引をアプリケーションプログ
 ラムから安全に伝送する方法及び装置に関する。

【0002】

【従来の技術】今日、クライアント／サーバ計算環境に
 おいては、オンライン金融取引を支援する大きい要求が
 存在している。これらのシステムを成功に導くキーは、
 許可されたパーティが許可された受信者へコミュニケーション
 を安全に伝送することを保証するセキュリティ機能
 である。最低でも、5つのセキュリティ機能、即ち、
 プライバシ、データの完全性、アクセス制御、ユーザ非
 拒絶、サーバ側監査証跡が必要である。金融サーバを呼
 出すクライアントのための現在のセキュリティ技術は、
 (A) クライアントのアイデンティティを指示するパス
 ワード及び類似物を使用するシステム、(B) 部外者が
 盗用することができないように、クライアントとサーバ
 との間のコミュニケーションを暗号化するセッションキ
 ーを使用するシステム、及び(C) デジタル署名及び
 証明プロセスを使用するシステムを含む。

【0003】セッションキーはプライバシー保護を与え、
 パスワードメカニズムは基本的なアクセス制御能力を与
 える。従来の技術は、クライアントであることを主張す
 るパーティに対して、実際に識別されたクライアント
 (または、そのパーティがクライアントのワークステ
 ーションを使用中であっても)であることを絶対的に保証
 せず、また特定のメッセージまたは要求を送らないク
 ライアントによる請求から金融機関を保護もしない。従来
 のシステムの監査証跡は、ログインするためのクライ
 アントのパスワードを使用してメッセージまたは要求を送
 ったパーティを示すだけであり、これは決定的な証拠に
 ならないことが多い。従って、クライアントが取引を要
 求しても金融機関が拒絶する可能性がある。更にコミュ
 ニケーションを暗号化するシステム(例えば、RSA暗
 号化ソフトウェアを使用する Quicken、Netscape、及び
 Schwab's Smart Money)は、各システムのソフトウェア

のTCP/IPプロトコル層では、そのように働く。こ
 の型のセキュリティ技術は、提供できるセキュリティ機
 能の型を制限する。例えば、SSLのようなTCP/IP
 プロトコルレベルにおけるセキュリティ機能は、典型
 的には、伝送される全てのデータを暗号化することによ
 ってプライバシーだけは与えるが、クライアントを認証
 することはできない。

【0004】クライアントデジタル署名を使用するシ
 ステムは、典型的にはデジタル証明書と共に使用され
 る。デジタル証明書は、信託された第三者によって署
 名された公開キーを必要とする。これらは、典型的に
 は、特定の公開キーが本当に特定のユーザの公開キーで
 あることを認証するために使用される。しかしながら、
 金融機関は詐欺的に悪用されて信頼を失うことに対して
 慎重であるために、デジタル証明書を使用することに
 乗り気ではない。

【0005】

【発明の概要】本発明は、金融サーバと、ウェブブラ
 ウザに関連するユーザとの間のコミュニケーションを安全
 ならしめるシステム及び方法に関する。金融取引が、公
 衆網を通してユーザとサーバとの間で安全に伝送される
 ように、コミュニケーションメカニズムが使用される。
 システムは、インターネットのようなコンピュータネッ
 トワークによって、サーバコンピュータに関連付けられ
 た少なくとも1つの金融サーバと相互に接続されている
 クライアントコンピュータに関連付けられたユーザのグ
 ループを含んでいる。金融サーバはウェブサーバを有
 し、このウェブサーバは、ユーザと、それらのウェブブ
 ラウザを通しての金融サーバとの間の対話を管理する。
 ウェブサーバは、金融サーバによって提供される種々の
 金融サービスに関連付けられたウェブページのリポジト
 リを有している。ウェブページは、ユーザとサーバとの
 間に交換された金融取引を表すHTML文書及びフォー
 ムを含む。ユーザは、ウェブブラウザを使用してHTML
 文書にアクセスし、HTMLからデータを戻す。次い
 でサーバは取引を処理し、各取引を追跡している監査証
 跡を更新する。

【0006】取引が高度に機密の性質であるために、本
 発明のシステム及び方法は、ウェブブラウザ及びウェブ
 サーバ内に幾つかのセキュリティ機能を組み入れている。
 以下の5つのセキュリティ機能が設けられている。
 即ち、セッションキー暗号化の形状のプライバシー、デー
 タ暗号化の使用を通してのデータの完全性、パスワード
 メカニズムを介してのアクセス制御、デジタル署名及
 びタイムスタンプによるユーザ非拒絶、及びサーバ側監
 査証跡である。ウェブブラウザには、暗号化されたフォ
 ムを受信し、タイムスタンプが付加され、デジタル
 サインされ、そして暗号化された形状のデータを含むメ
 ヂージを伝送する能力が与えられている。ウェブブラ
 ウザは、1対の暗号化キー、好ましくはプライベート及

び公開キー対を各ユーザに提供する能力を有している。ウェブブラウザの初期化手順は、設置中にこれらのキーを生成する。これらのキーは暗号化されたフォーマットで格納され、ブラウザからだけアクセス可能である。プライベートキーは、そのように要求された時に、取引メッセージにデジタル「サイン」するために使用される。

【0007】ウェブブラウザには、HTMLフォームを解説し、HTMLフォームデータを暗号化し、デジタルサインし、そしてタイムスタンプを付加するためのランダムセッションキーを生成する能力も与えられている。更に、ウェブブラウザは、HTML拡張をFORMタグに翻訳することができる。FORMタグは、HTMLフォームが暗号化されていることを指定する他に、HTMLフォームデータを3つのフォーマット、即ち

(1) 暗号化フォーマット、(2) デジタルサインされ、タイムスタンプを有するフォーマット、または(3) 暗号化され、デジタルサインされ、タイムスタンプを有しているフォーマットの1つで戻り伝送することを要求する。暗号化されたフォームデータを戻すために、ウェブブラウザは、フォームデータを含むメッセージを解説するのに使用されるランダムセッションキーを生成する。このセッションキーは戻りメッセージに添付され、サーバの公開キーを用いて暗号化される。メッセージのトップにはヘッダーレコードが含まれ、そのフォームデータは暗号化されていることを指示するフラグを含んでいる。

【0008】もしフォームがユーザのデジタル署名を要求すれば、ウェブブラウザは戻されるフォームデータにユーザのプライベートキーを使用して「サイン」し、デジタルタイムスタンプを添付する。ウェブサーバは、ユーザの公開キーを使用してユーザのデジタル署名を認証する。デジタル署名及び暗号化を要求するフォームの場合には、ウェブブラウザはユーザのプライベートキーを用いてフォームデータにサインし、デジタルタイムスタンプを添付する。ウェブブラウザは、フォームデータを暗号化するのに使用されるランダムセッションキーを生成する。このランダムセッションキーは戻りメッセージに添付され、サーバの公開キーで暗号化される。ウェブサーバは、受信したメッセージに関連付けられたヘッダーレコードを読み、そのメッセージのフォーマットを決定する。ヘッダーレコードに組み入れられたフラグが、特定のフォーマットを指示する。ウェブサーバは、そのメッセージを相応に処理し、もし適用可能であればフォームデータ、ユーザのデジタル署名、及びタイムスタンプを用いて監査証跡を更新する。ユーザのデジタル署名を検証するために、始めに、ユーザの会計が確立される時にウェブサーバによってユーザ登録プロセスが遂行される。この登録プロセスは、潜在的なユーザの金融データを引き出すユーザ登録HTMLフォ

ームによって容易にされる。ユーザは、ウェブサーバに戻される登録フォームデータにその公開キーを添える。ウェブサーバはこのユーザの公開キーをデータベース内に格納し、その後の取引において埋め込まれ得るユーザのデジタル署名を検証するために使用する。

【0009】本発明の付加的な目的及び特色は、以下の添付図面に基づく詳細な説明及び特許請求の範囲から容易に明白になるであろう。

【0010】

【実施例】

概要

図1を参照する。本発明は、クライアントコンピュータ106に関連する多数のユーザと通信するサーバコンピュータ102に関連する少なくとも1つの金融サーバを含む分散金融取引処理システム100及び方法に関する。金融サーバ102は、銀行、保険会社、仲買企業、消費者信用組合、等々のような金融機関が提供する種々の金融サービスを提供する。金融サービスは、オンライントレーディングサービス、投資信託成果情報または金融ニュースを得るための手段、ユーザのポートフォリオ保持を監視するための手段、等々を含むことができる。

【0011】金融サーバ102に接続されているユーザは、金融サービスの何れか1つを要求することができる。あるサービスには1つまたはそれ以上のウェブページが関連しており、ユーザの要求でクライアントコンピュータ106へダウンロードされる。ウェブページは、サーバ102へ伝送される情報を引き出すために、またはサーバ102からユーザ106へ情報を提供するために使用されるHTMLフォームを含むHTML文書124を含むことができる。更に、金融サーバ102は、各取引を追跡する監査証跡を含む。若干の場合には、HTMLフォーム124は、暗号化フォーマットで、または特別なフォーマットを何も行わずに、ウェブブラウザ216へ伝送することができる。ウェブブラウザはフォームデータを含むメッセージ143を、暗号化されたフォーマットで、ユーザのデジタル署名及びタイムスタンプを含むフォーマットで、ユーザのデジタル署名及びタイムスタンプを含み、暗号化されたフォーマットで、または特別なフォーマットを何も行わずに、金融サーバ102へ戻す。ウェブブラウザ216は、インカミング文書のフォーマットを指示し、且つ戻されるデータのフォーマットを指定するHTML FORMタグへの特別な拡張を認識する能力を有している。代替実施例は、他のHTMLタグへの拡張を指定することができる。

【0012】更に、ウェブブラウザ216には、暗号化手順220、タイムスタンプ手順228、デジタル署名手順230、及びランダムキー生成手順232も設けられている。ランダムキー生成手順232は、戻りメッセージを暗号化するセッションキーを生成するために暗

10

20

30

40

50

号化手順220と共に使用される。デジタル署名手順230及びタイムスタンプ手順228は、ウェブブラウザが戻りメッセージ143にデジタルサインし、タイムスタンプを添付することを可能にする。更に初期化手順226は、ユーザのデジタル署名を表し、且つ検証するために使用される暗号化キー218をウェブブラウザ216が生成できるようにする。

【0013】システムアーキテクチャ

図2に本発明の実施例による分散コンピュータシステム100を示す。システム100は、1つまたはそれ以上のサーバコンピュータ102と、1つまたはそれ以上のクライアントコンピュータ106とを含み、これらはコミュニケーションリンク104を通して互いに通信し合う。好ましくは、コミュニケーションリンク104は、インターネットのような公衆網である。サーバコンピュータ102は、中央処理ユニット(CPU)108、ユーザインタフェース110、コミュニケーションインタフェース112、及び主メモリ114を含む。コミュニケーションインタフェース112は、他のユーザワークステーション、並びに本発明には無関係な他のシステム資源と通信するために使用される。

【0014】サーバコンピュータ102の主メモリ114は、RAM(ランダムアクセスメモリ)、またはRAMと磁気ディスク記憶装置のような非揮発性メモリとの組合わせとして実現することができる。サーバコンピュータ102の主メモリ114は、以下のものを含むことができる。

- *オペレーティングシステム116、
 - *インターネットアクセス手順118、
 - *ウェブサーバ手順120、
 - *金融サーバ102により処理された各金融取引を追跡する監査証跡122、
 - *HTML文書リポジトリ124、
 - *データを暗号化し、解読するための暗号化手順126、
 - *デジタル署名をサインし、検証するためのデジタル署名手順128、
 - *各ユーザに関連する情報を格納するユーザデータベース132、
 - *サーバが使用するための1つまたはそれ以上の暗号化キー142、
 - *メッセージ143、並びに
 - *他のデータ構造及び手順。
- ユーザデータベース132は、以下のものを格納することができる。
- *会計識別子134、
 - *パスワード136、
 - *ユーザのデジタル署名を認証するために使用される1つまたはそれ以上のユーザの暗号化キー140、並びに

*他のデータ構造及び手順。

【0015】図3を参照する。クライアントコンピュータ106は、中央処理ユニット(CPU)202、ユーザインタフェース204、コミュニケーションインタフェース206、及び主メモリ208を含む。コミュニケーションインタフェース206は、他のユーザワークステーション、並びに本発明には無関係な他のシステム資源と通信するために使用される。クライアントコンピュータ106の主メモリ208は、RAM(ランダムアクセスメモリ)、またはRAMと磁気ディスク記憶装置のような非揮発性メモリとの組合わせとして実現することができる。サーバコンピュータ102の主メモリ208は、以下のものを含むことができる。

- *オペレーティングシステム210、
- *ネットワークアクセス手順212、
- *HTML文書リポジトリ214、
- *ウェブブラウザ216、
- *メッセージ143、並びに
- *他のデータ構造及び手順。

【0016】ウェブブラウザ216は、以下のものを含むことができる。

- *ユーザのデジタル署名に関連する1つまたはそれ以上のユーザ暗号化キー218、
- *データを暗号化し、解読するための暗号化手順220、
- *セッションキーをランダムに生成するためのランダムキー生成手順222、
- *戻りメッセージを、要求された手法で構成するためのフォーマッティング手順224、
- *ユーザの暗号化キー218を確立する初期化手順226、
- *時刻及び日付を表すタイムスタンプを生成するタイムスタンプ手順228、
- *あるフォームにサインし、またユーザのデジタル署名を検証するためのデジタル署名手順230、
- *ユーザのパスワード232、
- *ブラウザウェルカムウェブページ234
- *ユーザの暗号化キー218を暗号化するために使用されるブラウザの暗号化キー240、
- *サーバへの戻りメッセージを暗号化するために使用される1つまたはそれ以上の暗号化キー241、並びに
- *他のデータ構造及び手順。

【0017】ファイルフォーマット

本発明は、監査証跡、及び規定のフォーマットを有するフォームファイルを使用する。図4は、サーバの監査証跡122のフォーマットを表している。監査証跡122は、金融サーバ102によって処理された各取引を追跡するために使用される。監査証跡122はいろいろな目的のために役立つ。このような目的の1つは、クライアントメッセージ及び要求の検証である。クライアントに

よって開始され、金融サーバ102によって受信される各コミュニケーションは、監査証跡122内に記録される。監査証跡122内の各エントリ242は、特定のユーザに関連する会計識別子243、特定のユーザに関連するデジタル署名244、ユーザが取引にデジタルサインした日付と時刻を表すタイムスタンプ245、及びその取引に関連するテキスト246を含むことができる。若干の場合には、取引フォームは、ユーザのデジタル署名及びタイムスタンプを含む。これらの場合、ユーザのデジタル署名及びタイムスタンプは、ユーザが拒絶を主張する可能性に対して異議を唱えるために使用することができる。ユーザのデジタル署名及びタイムスタンプの存在は、特定のユーザがある日付及び時刻にその取引を実行したことを知らせるものである。拒絶主張に対して異議を唱える能力を「非拒絶」と呼ぶ。

【0018】本発明の便益は、詐欺及び誤りを減少させるのを3つの方法で援助することである。第1に、本発明は無許可のユーザが上述した取引を行うことを阻止する。第2に、実際に行われた取引をユーザが拒絶することを困難にする。最後に、本発明のセキュリティ機能は、インターネットを通してセキュリティ及び金融取引サービスを提供する Charles Schwab のような会社へ便益を与える。以上の説明から、監査証跡内に格納されているタイムスタンプは、実際にクライアントによって生成されることに注目されたい。構成の問題のために（例えば、ユーザのPCの時計が誤ってセットされている）、またはユーザの側に騙す意図があるために、このタイムスタンプが誤っている可能性がある。従って、監査証跡内に格納されているタイムスタンプは、単にその取引が行われた時点に関するクライアントの確認に過ぎない。サーバは、監査証跡エントリの信用性を増大させることができる幾つかの使用可能なオプションを有している。サーバはそれ自身のタイムスタンプを、監査証跡内のクライアントが供給したタイムスタンプと共に格納することができる。または、サーバは、タイムスタンプが現在の実際の時刻と大幅に異なる取引を拒絶することができる。サーバにとっての別の可能性は、タイムスタンプを生成し、それをHTML FORMタグへの別の拡張でクライアントへ送ることである。クライアントは、戻りメッセージの一部としてサーバが生成したタイムスタンプを含ませることになる。サーバにとっての更に別の可能性は、タイムスタンプを生成し、それをクライアントへ送る前にそれにデジタルサインすることである。これらの可能な実施例の全ては、監査証跡内のタイムスタンプの信用性を増大させ、従って詐欺及び誤りを減少させる上での監査証跡の有用性を向上させる。

【0019】本発明は、確認（オーセンティケーション）証明書を使用せずに、デジタル署名及び暗号化を使用する。確認証明書を使用せず、デジタル署名及び暗号化を使用すると、確認証明書を使用する従来技術に比し

て幾つかの長所が得られる。確認証明書インフラストラクチャをセットアップするのは、極めて時間がかかり且つ費用がかさむ。証明書を発行するプロセスは、極めて厄介である。確認証明書は、ある取引内に2つより多くのパーティが含まれている場合に有用である。本発明に使用される技術は、金融機関とその顧客の1つとの間のように2パーティ取引の場合に有効である。更に、これらの技術は、確認証明書インフラストラクチャのセットアップに要する費用を排除し、また証明書発行及び維持のオーバーヘッドを回避する。

【0020】図5及び6は、ウェブブラウザがメッセージ143をサーバ102へ戻すために使用される戻りメッセージレイアウトを示している。戻りメッセージ143は、会計識別子（ID）フィールド257、フラグフィールド258、及びキー長フィールド260を含むヘッダーレコード252を含んでいる。フラグフィールドは次のような値を取ることができる。即ち、“E”は暗号化されたフォーマットを表し、“S”はフォームデータ256がデジタル署名及びタイムスタンプを含んでいることを指示し、“A”は暗号化されたフォーマットがデジタル署名及びタイムスタンプを含んでいることを表し、そして“N”は特別なフォーマットではないことを表している。キー長フィールド260は、同封されている場合にはランダムセッションキーの長さを指示する。

【0021】戻りメッセージ143の第2のレコードは、ランダムセッションキー254であることができる。このキー254は、フォームデータ256が暗号化されていれば必ず同封される。データが暗号化されたフォーマットでなく伝送される場合には、セッションキーはフォームファイル内に含まれない。戻りメッセージ143の第3の部分は、フォームデータ256を含む。フォームデータ256は、ヘッダーレコード252のフラグフィールド258に示したようにフォーマットされる。本発明が、上述したようなファイルフォーマット及び内容に限定されるものではないことに注目すべきである。

【0022】HTML拡張

本発明は、HTML FORMタグへの拡張を使用する。拡張の1つのセットは、ユーザ登録HTMLフォーム264を処理することに関する。このフォームは、サーバの公開キーをユーザへ伝送し、またユーザがユーザの公開キーをサーバへ伝送するために使用される。拡張の第2のセットは、送受されるHTMLフォーム及び戻りメッセージのフォーマットを指示するために使用される。図7は、ユーザ登録フォーム268に関連するHTML FORMタグへの拡張を示す。第1は要求フィールド（例えば、request = “register”）であり、これは後続するフォームの型を指示するために使用される。「レジスタ」の値はそのフォームが、登録フォームであ

13

ることを指示している。第2の新しいフィールドはキーフィールド（例えば、key = "server's public key"）であり、これはサーバの公開キーを指示するために使用される。サーバの公開キーは、戻されるユーザ登録メッセージに関連する暗号化プロセスに使用される。図8は、インカミング（即ち、ウェブブラウザによって受信される）フォームまたはアウトゴーイング（即ち、ウェブブラウザからサーバへ伝送される）メッセージのフォーマットを指示するために使用される。HTML FORMタグは「アウトフォーマット」及び「インフォーマット」フィールドを含むことができる。アウトフォーマットフィールドは戻りメッセージのフォーマットを指定し、インフォーマットフィールドはインカミングの、即ち受信されるフォームのフォーマットを指定する。インフォーマットフィールドの値は、FORMタグに関連するインカミングフォームが暗号化されていることを指示する「暗号化」を含むことができる。アウトフォーマットフィールドの値は、「暗号化」、「サイン」、または「エンサイン」を含むことができる。「暗号化」値は、フォームデータが暗号化されたフォーマットで戻されることを通知する。「サイン」値は、戻されたフォームデータがユーザのデジタル署名及びタイムスタンプを含んでいることを通知する。「エンサイン」値は、戻されたフォームデータが暗号化され、ユーザのデジタル署名及びタイムスタンプを含んでいることを通知する。

【0023】更に、サーバの暗号化キーを情報及びアウトフォーマットフィールドと共に伝送するために、キーフィールドを使用することができる。公開キーを使用してセッションキーを暗号化するために、サーバの公開キーはウェブブラウザへ伝送される。一実施例では、サーバの公開キーは、暗号化を要求する各HTMLフォーム内に含ませることができる。別の実施例では、サーバの公開キーは、ユーザへ伝送される第1のHTMLフォーム内に含ませることができる。セッションが持続している間、それはウェブブラウザによって保持される。これは、その後の各伝送時にサーバの公開キーを再伝送する必要性を排除する。金融サーバから発するHTMLフォームは、FORMタグ内の上述した新しいフィールドのどのような組合わせをも含むことができる。更に、本発明は、如何なる新しいフィールドをも組み入れてないHTMLのバージョンに順応することもできる。

【0024】本発明をHTML FORMタグ内の新しいフィールドに関して特定の構文を参照して説明したが、この特定の実施例に限定されるものではない。類似の機能を提供するものであれば、他のどのような構文も使用することができる。更に、本発明は異なるHTMLタグを使用して、または新しいHTMLタグの使用を通して達成することができる。以上に本発明に関連する一般的なアーキテクチャに関して説明した。以下に本発明のアーキテクチャ、本発明によって遂行される処理、ア

14

ーキテクチャの要素間の区別、及び開示する技術に関連する長所をより詳細に説明する。

【0025】金融取引処理の概要

図9は、金融取引処理システム100及び本発明の方法に使用されるステップを示している。始めに、クライアントコンピュータ106に関連するユーザは、1対の暗号化キー218を生成するウェブブラウザ216を設置する（ステップ226）。好ましくは、プライベート／公開キー対218を作成し、ユーザがプライベートキーを使用してフォームにデジタルサインし、サーバが公開キーを使用してユーザのデジタル署名を認証する。ユーザは、金融サーバ102との会計134を確立する。この会計134は、始めにユーザが金融サーバ102にアクセスした時にユーザへ伝送されるユーザ登録HTMLフォーム264を通して確立される（ステップ274）。ユーザ登録HTMLフォーム264は、一般的なデータ、及び金融データ、並びにユーザの公開キー140をユーザから引き出す。一旦ユーザが金融サーバ102との会計134を確立すると、ユーザは種々の時点に会計134にログオンまたはアクセスすることができる（ステップ276）。各ログオンセッション中に、ユーザ及び金融サーバ102は、HTML取引フォーム124及びフォームデータ143を交換することができる（ステップ278）。フォームのフォーマットは、取引の型に依存して変化する。若干の場合には、サーバ102は暗号化されたフォームをユーザへ伝送することができる。他の場合には、サーバ102は、ウェブブラウザ216が複数のフォーマットでメッセージを戻すように要求することができる。特定のセッションが完了するとユーザは退出する（ステップ280）。後刻、ユーザは金融サーバ102と別のセッションを再活動させることができる。上述した各プロセスを以下に詳述する。

【0026】ウェブブラウザ初期化手順

先ず、金融取引システム100が始動する前に、ユーザはクライアントコンピュータ106内にウェブブラウザ216を設置する。ウェブブラウザの設置プロセスの一部として、初期化手順226が実行され、1つまたはそれ以上の暗号化キー218が生成される。好ましくは、プライベート及び公開キー対を生成する。プライベートキーはユーザのデジタル署名を作成するために使用され、公開キーはユーザのデジタル署名を検証するために使用される。図10及び11を参照する。初期化手順226はブラウザウェルカムページ234を表示する（ステップ300）。初期化手順226はブラウザ216に関連するパスワード232をユーザに促し、それを検証する（ステップ302）。検証に合格するとユーザのプライベート／公開暗号化キー218が生成され（ステップ304）、オプションでユーザに表示される（ステップ306）。公知のどのような暗号化技術も、暗号化キー218を生成するために使用することができる。

暗号化技術は、Schneier, Applied Cryptography, Jhon Wiley & Sons, 2ded., 1996 に記述されており、背景情報として本明細書に採り入れている。次いで初期化手順226は、ブラウザの暗号化キー240を使用してユーザの暗号化キー218を暗号化し(ステップ308)、これらのキーが不正に使用されたり、または悪用されないように保護し、それらをウェブブラウザアドレス空間内の所定の位置に格納する(ステップ310)。

【0027】金融サーバ処理

図12A-12Bは、金融サーバ102がユーザからの要求及び取引を処理するために使用するステップを示している。ユーザは、会計識別子134及びパスワード136によって金融サーバ102にアクセスする(ステップ320)。金融サーバ102は会計134及びそのパスワード136を検証する(ステップ322)。この検証は、会計134及びパスワード136と、ユーザデータベース132内のデータとを突き合わせることにによって遂行することができる。もし金融サーバ102が、そのユーザは新規であると決定すれば(ステップ324のY)、金融サーバ102はユーザ登録フォーム264をユーザへ伝送する(ステップ326)。ユーザ登録手順の詳細に関しては、図13を参照して後述する。ユーザが新規でなければ(ステップ324のN)、ユーザは金融サーバ102との会計134を確立する。金融サーバ102はクライアントからの伝送を待機する(ステップ328)。これらの伝送は、ウェブページ、フォームデータ、及び他の型のコミュニケーションに対する要求であることができる。金融サーバ102はクライアントから受信した伝送の型を識別し(ステップ330)、それを相応に処理する。

【0028】もし伝送が1つまたはそれ以上のウェブページに対する要求であれば、金融サーバ102は要求されたウェブページを入手し、それらをユーザへ伝送する(ステップ334)。ウェブページは暗号化されたHTMLフォーム(例えば、機密情報を有する文書が暗号化されている)を含むことができる。暗号化されたフォームは、図7を参照して前述したように、FORMタグ内の特別なフィールドによって識別される。クライアントからサーバへ送られる機密のユーザ情報を伴うメッセージは、好ましくはセッションキーを用いて暗号化する。サーバからクライアントへ送られるフォームは、好ましくはクライアントの公開キーを用いて暗号化する。これは、クライアントの公開キーで暗号化されるどのフォームまたは文書も、そのクライアントのプライベートキーにアクセスを有する個人またはシステムだけが解読し、見ることができるからである。

【0029】代替として、サーバからクライアントへ送られる機密情報を含むフォームは、サーバまたはクライアントの何れかが生成したセッションキーを用いて暗号化することができる。もしセッションキーがサーバによ

って生成されれば、そのセッションキーはクライアントの公開キーによって暗号化され、暗号化されたフォームに添付されてクライアントへ送られる。この実施例では、クライアントはまずそのプライベートキーを用いてセッションキーを解読し、次いでセッションキーを用いてフォームを解読する。典型的にセッションキー暗号化は、公開/プライベートキー暗号化よりも計算的に遙かに少ない費用である暗号化技術(DESのような)を使用して遂行されるので、文書を暗号化するには(短いメッセージとは対照的に)セッションキー暗号化が好まれることが多い。

【0030】更に別の代替実施例では、サーバは、各ユーザ毎に異なる対の公開/プライベート暗号化キーを使用することができる。別の代替実施例では、サーバは、各ログオンセッション毎に、そして各ユーザ毎に異なる対の公開/プライベート暗号化キーを使用することができる。もし伝送が戻りメッセージであれば、そのメッセージに関連するヘッダーレコード252のフラグフィールド258が解読され、フォーマットの型が識別される(ステップ332)。暗号化されたメッセージ(即ち、flag="E")の場合には、サーバのプライベートキー142を使用して暗号化されたセッションキー254を解読する(ステップ336)。セッションキーは、サーバの公開キーを用いてユーザのウェブブラウザ216によって暗号化されている。次いでセッションキー254を使用して同封されたフォームデータ256を解読する(ステップ338)。次にユーザの会計及びフォームデータ256を用いてサーバの監査証跡122が更新される(ステップ340)。次いでフォームデータ256が相応に処理される(ステップ342)。

【0031】ユーザのデジタル署名及びタイムスタンプを含む暗号化されたメッセージ(即ち、flag="A")の場合、埋め込まれたセッションキー254をサーバのプライベートキー142を使用して解読する(ステップ344)。次いで、このセッションキー254を使用して同封されたフォームデータ256を解読する(ステップ346)。次に、ユーザのデジタル署名218をメッセージ内で探知し、ユーザの公開キー140を用いて検証する。ユーザのデジタル署名に関連するタイムスタンプも、メッセージ内の所定の位置から抽出する(ステップ348)。ユーザの公開キー140を、サーバのユーザデータベース132から入手する。金融サーバ102は、メッセージに関連するヘッダーレコードの会計IDフィールド257を使用してユーザデータベース132を探索する。次いでユーザの会計、デジタル署名、タイムスタンプ、及びフォームデータを用いて監査証跡122が更新される(ステップ350)。次いでフォームデータが相応に処理される(ステップ352)。これで、手順は新しいクライアントコミュニケーションを待機する。

【0032】ユーザのデジタル署名を含むインカミングメッセージ143（即ち、flag="S"）は、ユーザのデジタル署名及びタイムスタンプで暗号化されているメッセージと同一のステップの若干を使用して処理される。金融サーバ102はメッセージ内のユーザのデジタル署名を探知し、ユーザの公開キー140を用いて検証する（ステップ348）。次いで、ユーザの会計、デジタル署名、タイムスタンプ、及びフォームデータを用いて監査証跡122が更新される（ステップ350）。次いでフォームデータが相応に処理される（ステップ352）。これで、手順は新しいクライアントコミュニケーションを待機する。特別なフォーマットを用い

ないで受信されたメッセージ（即ち、暗号化されていない、平文フォーマットで受信されたメッセージ）の場合、監査証跡122はユーザの会計及びフォームデータ256を用いて更新される（ステップ350）。次いでフォームデータが相応に処理される（ステップ352）。これで、手順は新しいクライアントコミュニケーションを待機する。

【0033】サーバのユーザ登録手順

図13は、図14に示すユーザ登録フォーム264を処理するために、ウェブブラウザ216が使用するステップを示している。ウェブブラウザ216は、ユーザ登録フォーム264を含むHTML文書を金融サーバ102から受信する（ステップ360）。ユーザ登録フォーム264は、ユーザが埋めるデータエントリフィールドを含む（ステップ362）。ウェブブラウザは、（クライアントコンピュータのユーザによって供給された、及び／または、クライアントコンピュータ内の他の資源から使用可能にされた）登録フォームをユーザ情報内に挿入

する。典型的には、この情報は、ユーザ名、社会保障番号または等価な識別子、及び先にユーザによってサーバに関連する金融機関と確立された会計のための金融機関会計番号のようなユーザを独自に識別するデータを含む。

【0034】FORMタグの要求フィールドから、そのフォームをユーザ登録フォームとして識別した（即ち、request="register"）ウェブブラウザ216は、ユーザの公開暗号化キー218を入手し、それを戻りメッセージ内の所定の位置に配置する（ステップ364）。ユーザの公開キー218は、暗号化されたフォーマットで、ウェブブラウザのアドレス空間内の指定された位置に格納される。ユーザの暗号化キー218は、ブラウザの暗号化キー240を用いて暗号化されている。従って、ウェブブラウザは既知の位置からのキーを、それ自身の暗号化キー240を用いて暗号化する。次いでウェブブラウザ216はデジタル署名手順230を使用し、ユーザのプライベートキー218を用いてフォームデータにデジタルサインする（ステップ366）。次に、ウェブブラウザ216はランダムセッションキー生

成手順222を使用してランダムセッションキー254を生成する（ステップ368）。ランダムセッションキー254は、フォームデータを暗号化するために使用される（ステップ370）。セッションキー254はメッセージのトップに添付され、サーバの公開キー142を用いて暗号化される。サーバの公開キー142はFORMタグ内のキーフィールドを通して伝送される（即ち、key="server'spublic key"）（ステップ372）。次いで戻りメッセージがフォーマット手順224を使用してフォーマットされ、サーバへ伝送される（ステップ374）。

【0035】ランダムセッションキーは、ランダムプロセスによって生成されたランダムなビット列である。典型的には、キービットは信頼できるランダム源、または暗号的に安全な擬似ランダムビットジェネレータから生成される。公知のランダムシーケンスジェネレータの何れを使用しても差し支えない。これらの技術の詳細は図10及び11に関連して前述したSchneierに記述されている。サーバは、受信した登録メッセージ内の情報を解説し、検証する。もし情報が、金融機関において先に確立された会計に関連するユーザ情報のような所定の受容基準を満足していれば、ユーザ情報レコードがユーザデータベース132に追加される。図2に示すように、ユーザ情報レコードは、ユーザ及びユーザの公開キーを識別する。代替として（例えば、もしサーバのデータベース132が既に関連金融機関に会計を有する各ユーザ毎のユーザ記録を含んでいれば）、既存のユーザ情報レコードは、そのユーザの公開キーを含むように更新される。

【0036】ウェブブラウザ

一旦ユーザが金融サーバ102に登録されると、ユーザはウェブブラウザ216を通してHTML取引フォーム及び戻りメッセージを金融サーバ102と交換する。サーバからクライアントへ送られる少なくとも若干のHTML文書は、HTML FORMタグを含む。好ましい実施例によれば、FORMタグは、関連HTML文書がどのようにフォーマットされているか（例えば、それが暗号化されているか否か）を指示する特別フィールドを含む。もしサーバからクライアントへ送られるHTML文書が暗号化されていれば、FORMタグ内の特別な「キー」フィールドを使用してサーバの公開キーを指定することができる。もしフォームが、ユーザ情報をサーバへ戻すことを要求する型のものであれば、FORMタグは、戻りメッセージをサーバに戻す前に、それをどのようにフォーマットすべきかをクライアントのウェブブラウザに指令する特別なフィールドをも含む。クライアントのウェブブラウザ216はこれらのFORMタグフィールドを読み、ユーザがHTML文書を読んでサーバへ戻り伝送するメッセージを適切にフォーマットできるようにする適切な手順を遂行する。

19

【0037】図15A-15Bは、金融サーバ102から受信したHTML文書処理するためにウェブブラウザ216が使用するステップを示している。HTML文書を受信すると（ステップ380）、ウェブブラウザ216はFORMタグのフィールドを調べる。もしFORMタグが、そのタグに関連するフォームが暗号化されていることを指示していれば（即ち、インフォーマットフィールドが存在）（ステップ382のY）、FORMタグ対の間に含まれるデータが暗号化されていることをウェブブラウザ216が認識する。ウェブブラウザ216は、対応するFORMタグ対に到達するまで（即ち、`</FORM>`）ファイルからデータを読み取る（ステップ414）。ウェブブラウザ216は、ユーザのプライベートキー218を用いてフォームを解読し（ステップ416）、HTML文書内のタグを読み取り続ける。

【0038】次に、ウェブブラウザ216はFORMタグがアウトフォーマットフィールドを有しているか否かを検出する（ステップ384）。もしFORMタグがアウトフォーマットフィールドを有していなければ（ステップ384のN）、フォームが表示され、相応に処理される（ステップ386）。もしFORMタグがアウトフォーマットフィールドを有していれば（ステップ384のY）、アウトフォーマットフィールドは格納され、フォームが表示され、相応に処理される（ステップ388）。一旦フォームが処理されてしまうと、ウェブブラウザ216は、格納されているアウトフォーマットフィールド内に指定されている要求された指示文に従って戻りメッセージを準備する。

【0039】もし要求された戻りメッセージフォーマットがデジタル署名及びタイムスタンプを用いて暗号化することを指定していれば（即ち、`outformat = "ensign"`）、ウェブブラウザ216はフォームデータをユーザのプライベートキー218を用いてデジタルサインし、メッセージ内の所定の位置にタイムスタンプを添付する（ステップ392）。次に、ウェブブラウザ216はセッションキー254をランダムに生成する（ステップ394）。次いでメッセージをこのランダムに生成したセッションキー254を用いて暗号化する（ステップ396）。若干の実施例では、単一のセッション中の全ての暗号化されたクライアントメッセージ伝送のために単一のセッションキーが使用され、この場合、あるセッション中に第1の暗号化されたクライアントメッセージの伝送の後に、ステップ396はスキップされる。

【0040】セッションキー254はサーバの公開キー142を用いて暗号化され、暗号化されたメッセージに添付される（ステップ398）。図8に関して説明したように、サーバの暗号化キーは、始めに第1の伝送されるHTMLフォームと共に、または伝送される各HTMLフォームと共にウェブブラウザに伝送されて、暗号化を要求する。次いで、適切な値（“A”）を有するフラ

20

グ258及び同封されるセッションキー254のキーの長さ260を含むヘッダーレコード252が生成される。メッセージがフォーマットされ、金融サーバ102へ伝送される（ステップ400）。もし要求された戻りメッセージフォーマットが暗号化を指定していれば（即ち、`outformat = "encrypt"`）、ウェブブラウザ216は上述したステップと同一のステップの若干を遂行する。ウェブブラウザ216はセッションキー254をランダムに生成する（ステップ394）。次いでフォームデータをこのランダムに生成したセッションキー254を用いて暗号化する（ステップ396）。セッションキー254は暗号化されたフォームデータに添付され、サーバの公開キー142を用いて暗号化される（ステップ398）。次いで適切な値（“E”）を有するフラグ258及び同封されるセッションキーのキーの長さ260を含むヘッダーレコード252が生成される。メッセージがフォーマットされ、金融サーバ102へ伝送される（ステップ400）。

【0041】もし要求された戻りメッセージフォーマットがユーザのデータ署名を指定していれば（即ち、`outformat = "sign"`）、ウェブブラウザ216はユーザのプライベートキー218を用いてフォームデータにサインする（ステップ410）。更に、タイムスタンプが生成され、フォームデータに添付される（ステップ410）。次いで適切なフラグ値（“S”）を含むヘッダーレコード252が生成される。メッセージ内にはセッションキーが同封されないから、キー長フィールドは空白である。メッセージがフォーマットされ、金融サーバ102へ伝送される（ステップ412）。結論として、以上の説明は、金融サーバとウェブブラウザとの間に、HTMLフォームとして実現される取引を安全に伝送するための方法及びシステムを記述している。本発明の技術は、意図された取引のパーティだけが取引を安全に送受信するように、5つのセキュリティ機能を組み入れている。5つのセキュリティ機能は、セッションキー暗号化の形状のプライバシ、データ暗号化の使用を通してのデータの完全性、パスワードメカニズムを介してのアクセス制御、デジタル署名及びタイムスタンプによるユーザ拒絶、及びサーバ側監査証跡を含む。これらのセキュリティ機能は、自動的な及び透明な手法で金融サーバ及びウェブブラウザ内に埋め込まれている。

【0042】代替実施例

以上に幾つかの実施例を参照して本発明を説明したが、この説明は本発明を例示したに過ぎず、本発明を限定する意図はない。当分野に精通していれば、特許請求の範囲に限定されている本発明の思想及び範囲から逸脱することなく多くの変更を考案できよう。更に、上述した方法及びシステムは、ランダムアクセスメモリのようなメモリデバイス以外の種々の型の実行可能な媒体で実行させるように改変することができる。限定するものではな

いが、メモリデバイス、コンパクトディスク、またはフロッピーディスクの何れであることもできるコンピュータ可読記憶媒体のような他の型の実行可能な媒体も使用することができる。

【0043】本発明を暗号化キー対を使用する暗号化及びデジタル署名技術に関連して説明したが、本発明はこの特定の技術に限定されるものではない。類似の機能を提供するどのような技術も使用することが可能である。更に、当分野に精通していれば、ウェブブラウザに伝送されるHTMLフォーム内にデジタル署名メカニズムを組み入れるように本発明を容易に変更できよう。更に、取引処理のクライアント側またはサーバ側の何れかに付加的なセキュリティ機能を容易に追加することが可能である。

【図面の簡単な説明】

【図1】本発明の実施例による金融取引処理システムを示す図である。

【図2】本発明の実施例によるサーバコンピュータシステムを示す図である。

【図3】本発明の実施例によるクライアントコンピュータシステムを示す図である。

【図4】サーバコンピュータ内に存在する監査証跡のフォーマットを示す図である。

【図5】本発明による戻りメッセージレイアウトを示す図である。

【図6】図5の戻りメッセージのヘッダーレコードのレコードレイアウトを示す図である。

【図7】ユーザ登録フォームと共に使用されるHTML FORMタグの例の概要図である。

【図8】インカミングフォーム及び戻りメッセージのフォーマットを指示する付加的なフィールドを含むHTML FORMタグの例の概要図である。

【図9】本発明の金融取引処理システムが使用するステップの流れ図である。

【図10】ユーザのための暗号化キーを確立するためにウェブブラウザが使用するステップの流れ図である。

【図11】ユーザの暗号化キーの生成を開始するためにウェブブラウザが使用するウェブページ例の概要図である。

【図12A】クライアントコンピュータに関連するユーザと通信するために金融サーバによって使用されるステップの流れ図である。

【図12B】クライアントコンピュータに関連するユーザと通信するために金融サーバによって使用されるステップの流れ図である。

【図13】ユーザ登録HTMLフォームを処理するためにウェブブラウザが使用するステップの流れ図である。

【図14】ユーザ登録HTMLフォーム例の概要図であ

る。

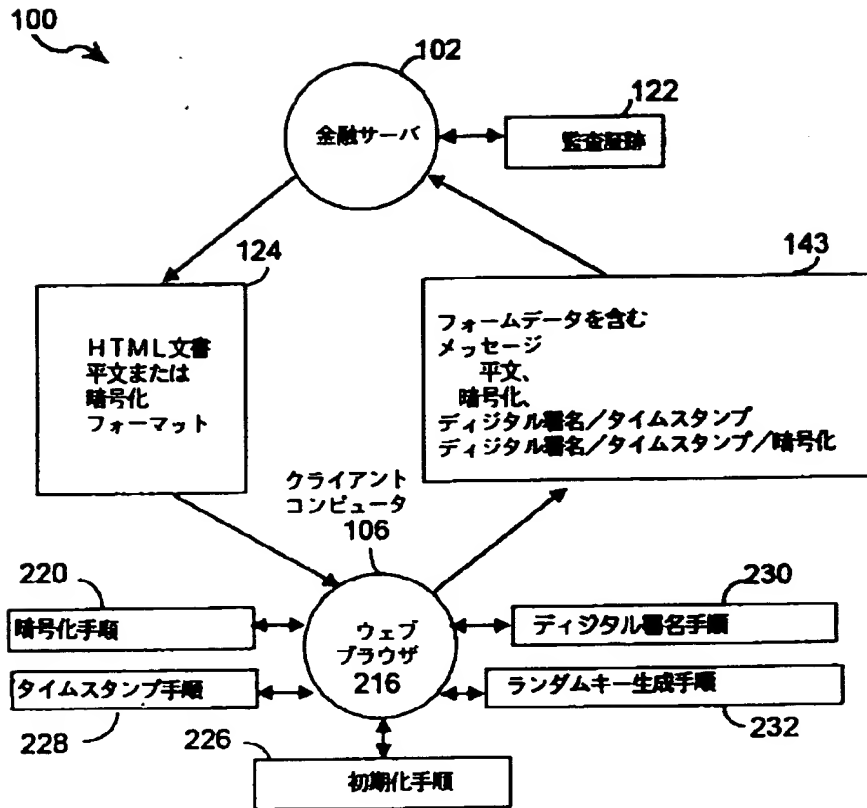
【図15A】HTML文書进行处理するためにウェブブラウザが使用するステップの流れ図である。

【図15B】HTML文書进行处理するためにウェブブラウザが使用するステップの流れ図である。

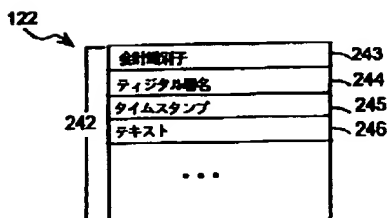
【符号の説明】

100 分散金融取引処理システム
102 サーバコンピュータ
104 コミュニケーションリンク
106 クライアントコンピュータ
108、220 CPU
110、204 ユーザインタフェース
112、206 コミュニケーションインタフェース
114、208 主メモリ
116、210 オペレーティングシステム
118、212 インターネット（ネットワーク）アクセス手順
120 ウェブサーバ手順
122 監査証跡
124、214 HTML文書
126、220 暗号化手順
128、230 デジタル署名手順
132 ユーザデータベース
134 会計識別子
136、232 パスワード
140、218 ユーザの暗号化キー
142 サーバの暗号化キー
143 戻りメッセージ
216 ウェブブラウザ
222、232 ランダムキー生成手順
224 フォーマット手順
226 初期化手順
228 タイムスタンプ手順
234 ブラウザウェルカムウェブページ
240 ブラウザの暗号化キー
241 暗号化キー
242 エントリ
243 会計識別子
244 デジタル署名
245 タイムスタンプ
246 テキスト
252 ヘッダーレコード
254 ランダムセッションキー
256 フォームデータ
257 会計識別子フィールド
258 フラグフィールド
260 キー長フィールド
264、268 ユーザ登録フォーム

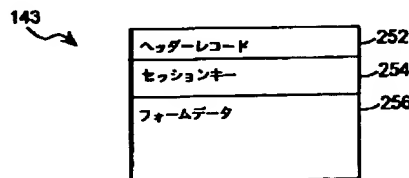
【図1】



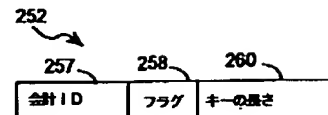
【図4】



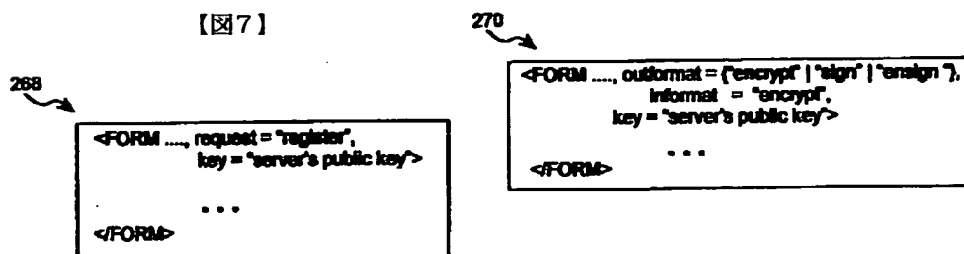
【図5】



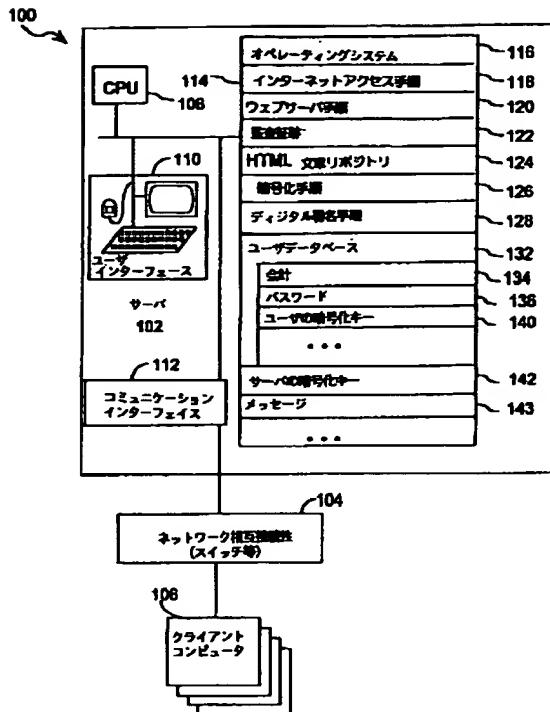
【図6】



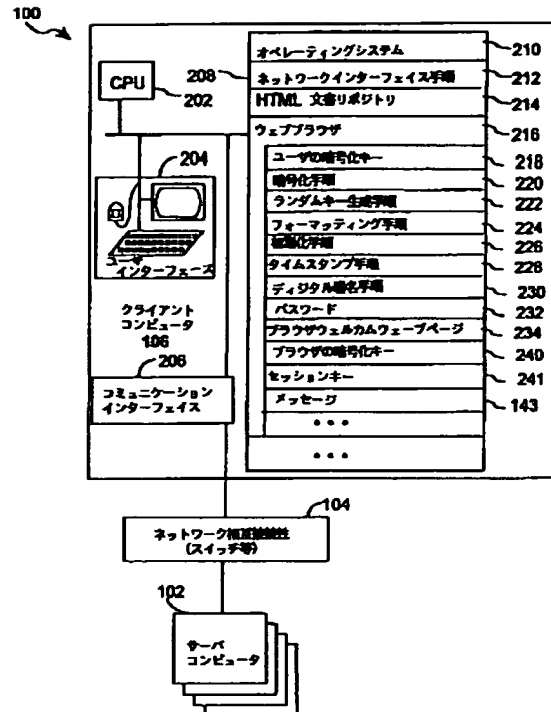
【図8】



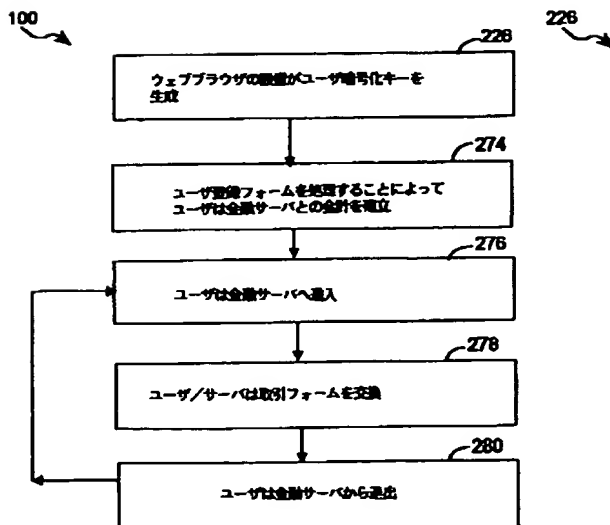
【図2】



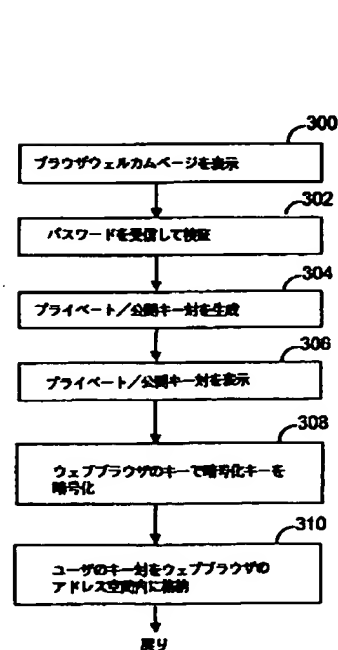
【図3】



【図9】



【図10】



【図11】

234 ↗

ブラウザウェルカムページ

暗号化キー対照立

パスワード: を入力

暗号キー:

プライベートキー

公開キー

【図14】

264 ↗

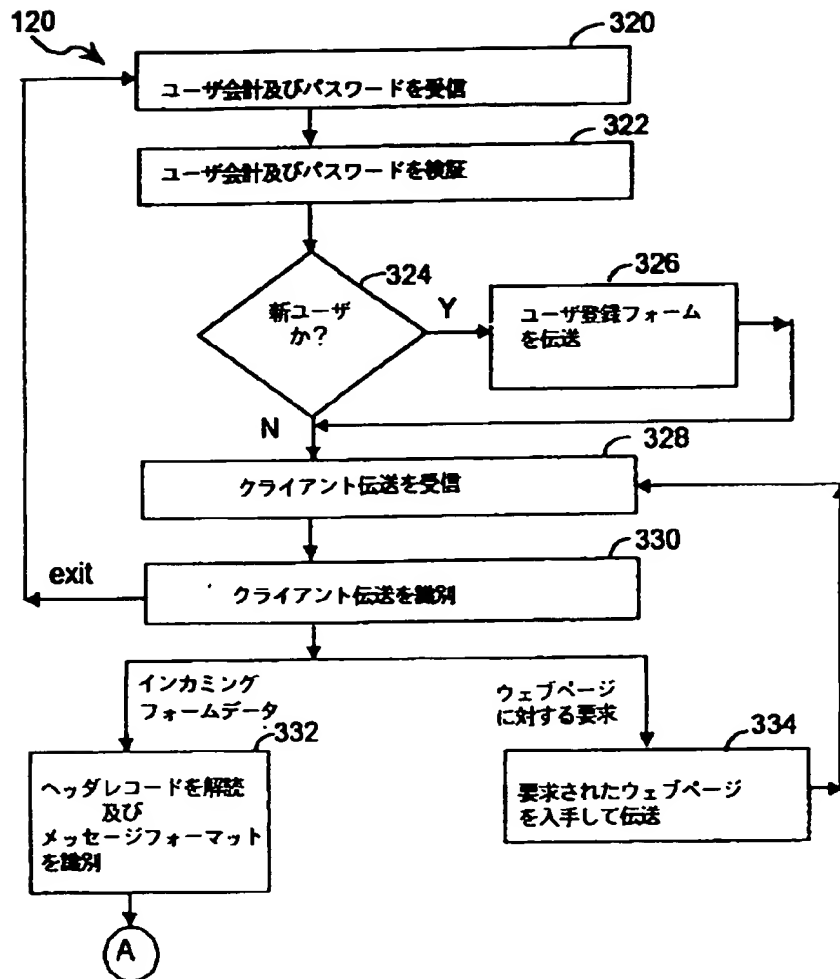
ユーザ登録フォーム

名前:

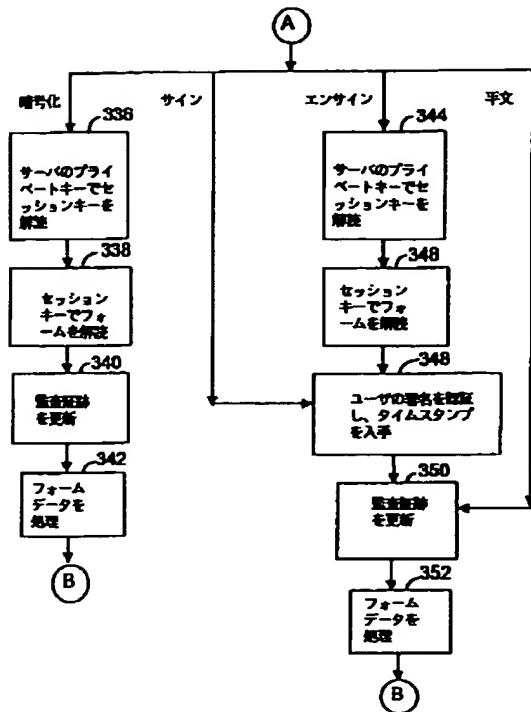
住所:

...

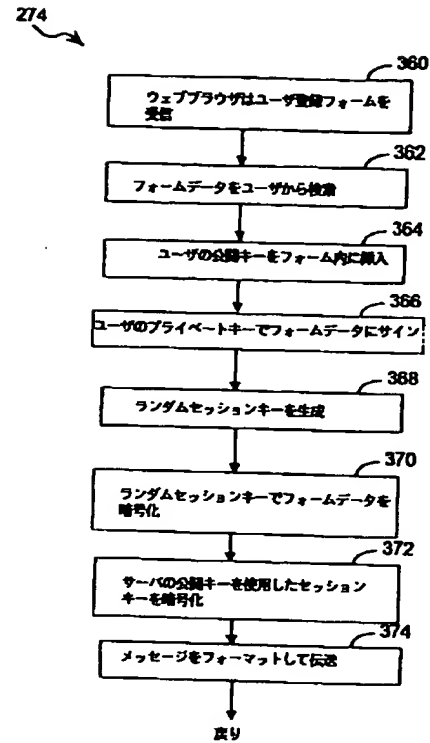
【図12A】



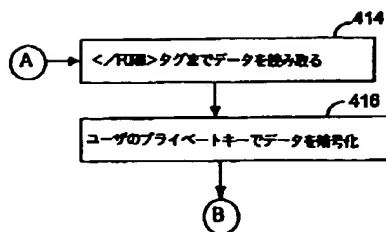
【図12B】



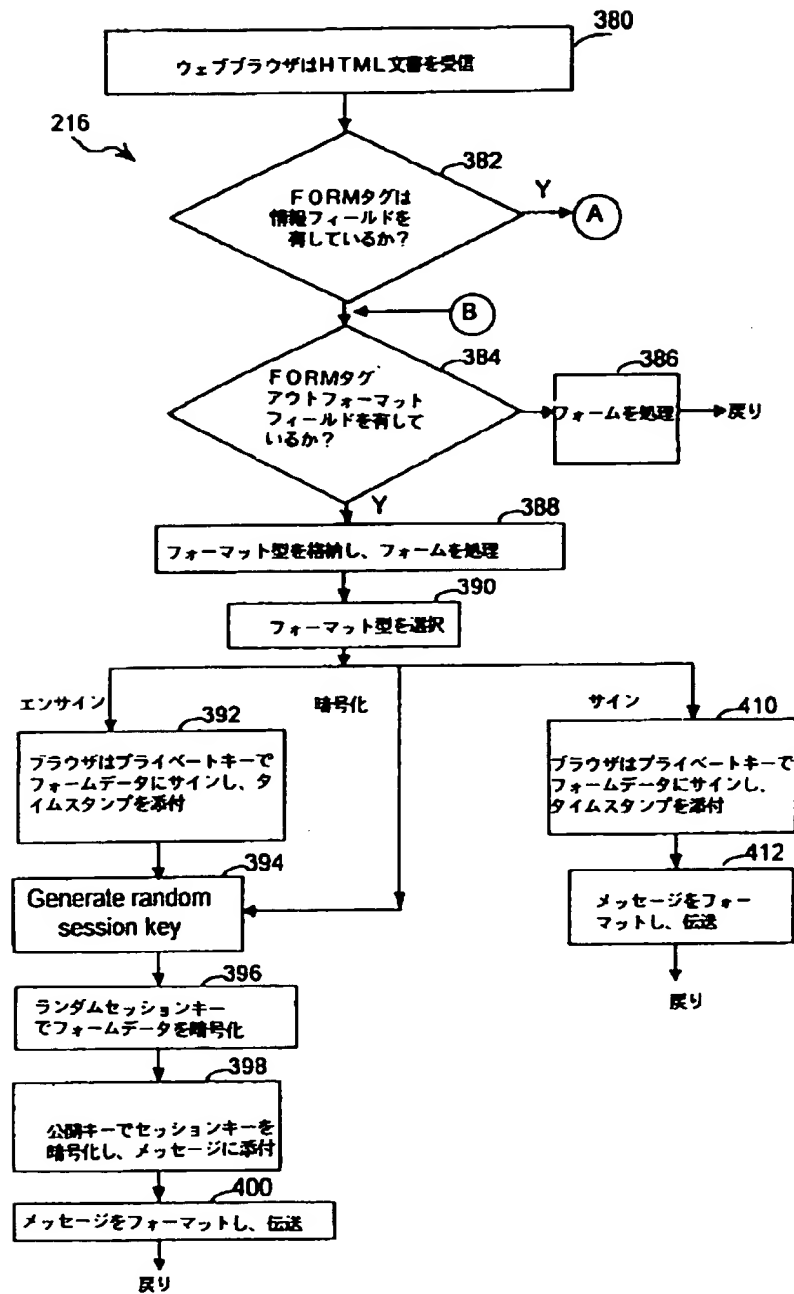
【図13】



【図15B】



【図15A】



フロントページの続き

(51)Int. Cl.⁶

G09C 1/00

識別記号

660

FI

G06F 15/21

15/30

Z

H

(71)出願人 591064003
901 SAN ANTONIO ROAD
PALO ALTO, CA 94303, U.
S. A.

(72)発明者 スチュアート マークス
アメリカ合衆国 カリフォルニア州
94043 マウンテン ヴィュー シエラ
ヴィスタ アベニュー 450-2

【外国語明細書】

SECURITY SYSTEM AND METHOD FOR FINANCIAL INSTITUTION SERVER AND CLIENT WEB BROWSER

This invention relates to electronic communication systems and in particular to a method and apparatus for securely transmitting transactions from an application program.

Background of the Invention

Today, there is a great demand to support online financial transactions in a client/server computing environment. The key to the success of such systems are security features to ensure that an authorized party securely transmits a communication to an authorized recipient. At a minimum, five security features are needed: privacy, data integrity, access control, user nonrepudiation, and a server side audit trail.

Current security technology for clients calling financial servers include (A) systems using passwords and the like to indicate a client's identity, (B) systems using session keys to encrypt the communications between a client and server so that outsiders cannot eavesdrop, and (C) systems using a digital signature and certification process.

Session key encryption provides privacy protection, and password mechanisms provide basic access control capabilities. The prior art does not provide absolute assurance that the party claiming to be a client is in fact the identified client (or even that the party is using the client's workstation), and also does not protect the financial institution from claims by clients that they did not send a particular message or

- 2 -

request. The audit trail of the prior art systems will only show that the party that sent the message or request used the client's password to log in, which is often not conclusive proof. Thus, the financial institution is at risk of repudiation of transactions by clients.

Furthermore, the systems (e.g., Quicken, Netscape and Schwab's Smart Money using RSA encryption software) for encrypting the communications do so at the TCP/IP protocol layer of each system's software. This type of security technology limits the type of security features that can be provided. For instance, a security feature at the TCP/IP protocol level, such as SSL, typically provides only privacy by encrypting all data transmitted, but it cannot authenticate the client.

Systems utilizing client digital signatures are typically used with digital certificates. Digital certificates require a public key to be signed by a trusted third party. They are typically used to authenticate that a particular public key is really that of a particular user. However, financial institutions are reluctant to utilize digital certificates since they are wary of the potential liability associated with their fraudulent misuse.

Summary of the Invention

The present invention pertains to a system and method for providing a secure communication mechanism between a financial server and a user associated with a web browser. The communication mechanism is used to ensure that financial transactions are securely transmitted between the user and server across a public network. The system includes a group of users associated with client computers that are interconnected, by a computer network such as the Internet, to at least one financial server associated with a server computer.

The financial server has a web server that manages the interactions between the users, through their web browsers, and the financial server. The web server has a repository of web pages associated with various financial services provided by the financial server. The web pages contain HTML documents and forms representing

financial transactions that are exchanged between the user and the server. A user utilizes a web browser to access the HTML documents and to return data from HTML forms to the server. The server then processes the transactions and updates an audit trail that tracks each transaction.

Due to the highly confidential nature of the transactions, the system and method of the present invention incorporates several security features into the web browser and web server. The following five security features are provided: privacy, in the form of session key encryption; data integrity, through the use of data encryption; access control, via a password mechanism; user nonrepudiation, by means of digital signatures and timestamps; and a server side audit trail.

The web browser is provided with the capability to receive encrypted forms and to transmit messages containing timestamped, digitally signed, and encrypted form data. The web browser has the ability to provide each user with a pair of encryption keys, preferably a private and public key pair. The web browser's initialization procedure generates these keys during installation. The keys are stored in an encrypted format and are only accessible from within the browser. The private key is used to digitally "sign" a transaction message when so requested.

The web browser is also provided with the capability to generate random session keys, to decrypt HTML forms, and to encrypt and digitally sign and timestamp HTML form data. In addition, the web browser can interpret HTML extensions to the FORM tag that specify that an HTML form is encrypted as well as request the return transmission of HTML form data in one of three formats: (1) encrypted; (2) digitally signed with a timestamp; or (3) encrypted and digitally signed with a timestamp.

To return the encrypted form data, the web browser generates a random session key that is used to encrypt the message containing the form data. The session key is affixed to the return message and encrypted with the server's public key. A header record is included at the top of the message and includes a flag indicating that the form data is encrypted.

- 4 -

If a form requires the user's digital signature, the web browser uses the user's private key to "sign" the returned form data and to affix a digital timestamp. The web server authenticates the user's digital signature with the user's public key. For forms requiring a digital signature and encryption, the web browser signs the form data with the user's private key and includes a digital timestamp as well. The web browser generates a random session key that is used to encrypt the form data. The random session key is affixed to the return message and encrypted with the server's public key.

The web server reads the header record associated with a received message in order to determine the format of the message. A flag associated with the header record indicates the particular format. The web server processes the message accordingly and updates an audit trail with the form data, the user's digital signature, and timestamp, if applicable.

In order to verify a user's digital signature, a user registration process is performed initially by the web server at the time a user's account is established. The registration process is facilitated by a user registration HTML form that elicits financial data pertaining to a potential user. A user appends its public key to the registration form data which is returned to the web server. The web server stores the user's public key in a database and it is used thereafter to verify the user's digital signature which may be embedded in subsequent transactions.

Brief Description of the Drawings

Additional objects and features of the invention will be more readily apparent from the following Detailed Description and appended claims when taken in conjunction with the drawings in which:

Fig. 1 shows a financial transaction processing system according to an embodiment of the present invention.

Fig. 2 shows a server computer system according to an embodiment of the present invention.

Fig. 3 shows a client computer system according to an embodiment of the present invention.

Fig. 4 shows a format of an audit trail residing in the server computer.

Fig. 5 shows a return message layout in accordance with the present invention.

Fig. 6 shows the record layout of the header record of the return message of Fig. 5.

Fig. 7 is a schematic representation of an exemplary HTML FORM tag used in connection with a user registration form.

Fig. 8 is a schematic representation of an exemplary HTML FORM tag including additional fields that indicate the format of incoming forms and return messages.

Fig. 9 is a flow chart of the steps used in the financial transaction processing system of the present invention.

Fig. 10 is a flow chart of the steps used by the web browser to establish encryption keys for a user.

Fig. 11 is a schematic representation of an exemplary web page used by the web browser to initiate the generation of a user's encryption keys.

Figs. 12A - 12B are flow charts of the steps used by the financial server in communicating with users associated with client computers.

Fig. 13 is a flow chart of the steps used by the web browser in processing a user registration HTML form.

- 6 -

Fig. 14 is a schematic representation of an exemplary user registration HTML form.

Figs. 15A - 15B are flow charts of the steps used by the web browser in processing HTML documents.

DESCRIPTION OF THE PREFERRED EMBODIMENTS

Overview

Referring to Fig. 1, the present invention pertains to a distributed financial transaction processing system 100 and method including at least one financial server associated with a server computer 102 in communication with a number of users associated with client computers 106.

The financial server 102 provides various financial services which are offered by a financial institution such as a bank, insurance company, brokerage firm, credit union, and the like. The financial services can include an online trading service, means for obtaining investment product information or financial news, means for monitoring a user's portfolio holding, and the like.

Users connected to the financial server 102 can request any one of the financial services. One or more web pages are associated with a service and are downloaded to the client computer 106 at the user's request. The web pages can include HTML documents 124 containing HTML forms used to elicit information from the user 106 that is transmitted to the server 102 or to provide information to the user 106 from the server 102. In addition, the financial server 102 contains an audit trail 122 that tracks each transaction.

In certain cases, the HTML forms 124a can be transmitted to the web browser 216 in an encrypted format or without any special formatting. The web browser returns messages 143 containing form data to the financial server 102 in an encrypted format, a format containing the user's digital signature and timestamp, an encrypted

- 7 -

format containing the user's digital signature and timestamp, or without any special formatting. The web browser 216 has the capability to recognize special extensions to the HTML FORM tag that indicate the format of the incoming document and specify the format of the data to be returned. An alternate embodiment might specify extensions to other HTML tags.

In addition, the web browser 216 is equipped with encryption procedures 220, timestamp procedures 228, digital signature procedures 230, and random key generation procedures 232. The random key generation procedures 232 are used to generate session keys that are used in conjunction with the encryption procedures 220 to encrypt a return message. The digital signature 230 and timestamp 228 procedures enable the web browser to digitally sign and timestamp a return message 143. In addition, the initialization procedures 226 enable the web browser 216 to generate encryption keys 218 that are used to represent and verify a user's digital signature.

System Architecture

Referring to Fig. 2, there is shown a distributed computer system 100 according to an embodiment of the present invention. The system 100 includes one or more server computers 102 and one or more client computers 106 that are in communication with each other through a communication link 104. Preferably, the communication link 104 is a public network such as the Internet.

A server computer 102 includes a central processing unit (CPU) 108, a user interface 110, a communications interface 112, and a primary memory 114. The communications interface 112 is used to communicate with other user workstations as well as other system resources not relevant here.

The primary memory 114 of the server computer 102 may be implemented as RAM (random access memory) or a combination of RAM and non-volatile memory such as magnetic disk storage. The primary memory 114 of the server computer 102 can contain the following:

- 8 -

- an operating system 116;
- Internet access procedures 118;
- Web server procedures 120;
- an audit trail 122 tracking each financial transaction processed by the financial server 102;
- an HTML document repository 124;
- an encryption procedure 126 for encrypting and decrypting data;
- a digital signature procedure 128 for signing and verifying a digital signature;
- a user database 132 storing information associated with each user;
- one or more encryption keys 142 for use by the server;
- messages 143;
- as well as other data structures and procedures.

The user database 132 can store the following:

- an account identifier 134;
- a password 136;
- one or more of the user's encryption keys 140 that are used to authenticate a user's digital signature; and
- as well as other data structures and procedures.

Referring to Fig. 3, a client computer 106 includes a central processing unit (CPU) 202, a user interface 204, a communications interface 206, and a primary memory 208. The communications interface 206 is used to communicate with other user workstations as well as other system resources not relevant here.

The primary memory 208 of the client computer 106 may be implemented as RAM (random access memory) or a combination of RAM and non-volatile memory such as magnetic disk storage. The primary memory 208 of the server computer 102 can contain the following:

- an operating system 210;
- network access procedures 212;
- an HTML document repository 214;

- 9 -

- a web browser 218;
- messages 143;
- as well as other data structures and procedures.

The web browser 216 can contain the following:

- one or more user encryption keys 218 associated with the user's digital signature;
- an encryption procedure 220 for encrypting and decrypting data;
- a random key generation procedure 222 for randomly generating session keys;
- a formatting procedure 224 for configuring a return message in a requested manner;
- an initialization procedure 226 that establishes the user's encryption keys 218;
- timestamp procedures 228 that generate a timestamp representing a time and date;
- digital signature procedures 230 that are used to sign a form and verify a user's digital signature;
- the user's password 232;
- a browser welcome web page 234;
- a browser's encryption key 240 that is used to encrypt the user's encryption keys 218;
- one or more session keys 241 for use in encrypting return messages to the server;
- as well as other data structures and procedures.

File Formats

The present invention utilizes an audit trail and a form file having prescribed formats. Fig. 4 represents a format of the server's audit trail 122. The audit trail 122 is used to track each transaction that is processed by the financial server 102. The audit trail 122 serves a variety of purposes. One such purpose is to verify client messages and requests. Each communication initiated by a client and received by the financial server 102 is recorded in the audit trail 122.

- 10 -

Each entry 242 in the audit trail 122 can contain an account identifier 243 associated with a particular user, the digital signature 244 associated with a particular user, a timestamp 245 representing the date and time at which the user digitally signed the transaction, and the text 246 associated with the transaction. In some cases, a transaction form will contain the user's digital signature and timestamp. In these cases, the user's digital signature and timestamp can be used to refute potential repudiation claims by the user. The existence of the user's digital signature and timestamp signifies that a particular user executed the transaction and at a certain date and time. The ability to refute repudiation claims is sometimes called "nonrepudiation."

The benefit of this invention is that it helps reduce fraud and errors in three ways. First, it prevents unauthorized users from forging transactions. Second, it makes it difficult for a user to repudiate a transaction that has actually been made. Lastly, audit trail entries are potentially useful as judicial evidence. Furthermore, the security features of this invention benefit companies such as Charles Schwab that offer securities and financial transaction services over the Internet.

It should be noted that, in the description above, the timestamp stored in the audit trail was actually generated by the client. It is possible for this timestamp to be in error, either because of a configuration problem (e.g., the user's PC had its clock set wrong) or because of fraudulent intent on the part of the user. The timestamp stored in the audit trail is thus merely the client's assertion as to the time at which the transaction was made. The server has several options available that can increase the trustworthiness of the audit trail entries. The server can store a timestamp of its own along with the client-supplied timestamp in the audit trail. Or, the server can reject transactions whose timestamps differ from the current actual time by a significant amount. Another possibility is for the server to generate a timestamp and send it to the client in another extension to the HTML FORM tag. The client would include the server-generated timestamp as part of the return message. Still another possibility would be for the server to generate a timestamp and to digitally sign it before sending it to the client. All of these possible embodiments increase the

trustworthiness of the timestamps in the audit trail, thus serving to increase the usefulness of the audit trail in reducing fraud and errors.

This invention employs digital signatures and encryption without using authentication certificates. Using digital signatures and encryption without authentication certificates offers several advantages over the prior art use of authentication certificates. Setting up the authentication certificate infrastructure is very time-consuming and costly. The process of issuing certificates is very cumbersome. Authentication certificates are useful in cases where more than two parties are involved in a transaction. The techniques employed in this invention are effective for two-party transactions, such as between a financial institution and one of its customers. Furthermore, these techniques avoid the expense of setting up the authentication certificate infrastructure as well as avoiding the overhead of certificate issuance and maintenance.

Figs. 5 and 6 illustrate the return message layout that the web browser utilizes in returning a message 143 back to the server 102. The return message 143 contains a header record 252 containing an account identification (ID) field 257, a flag field 258 and a key length field 260. The flag field can take on the following values: "E", representing an encrypted format; "S", indicating that the form data 256 contains a digital signature and a timestamp; "A", representing an encrypted format containing a digital signature and timestamp; and "N", representing no special formatting. The key length field 260 indicates the length of a random session key when enclosed.

The second record of the return message 143 can be the random session key 254. This key 254 is enclosed in the return message whenever the form data 256 is encrypted. In cases where the data is not transmitted in an encrypted format, no session key is included in the form file. The third portion of the return message 143 contains the form data 256. The form data 256 is formatted as indicated in the flag field 258 of the header record 252.

It should be noted that the present invention is not limited to the file formats and contents as previously described. Other formats can be used that can include different contents.

HTML Extensions

The present invention employs extensions to the HTML FORM tag. One set of extensions pertains to processing the user registration HTML form 264. This form is used to transmit the server's public key to the user and for the user to transmit to the server the user's public key. A second set of extensions is used to indicate the format of received and transmitted HTML forms and return messages.

Fig. 7 shows the extensions to the HTML FORM tag associated with a user registration form 268. The first is the request field (e.g., request = "register") which is used to indicate the type of form that follows. A value of *register* indicates that the form is a registration form. The second new field is the key field (e.g., key = "server's public key") which is used to indicate the server's public key. The server's public key is used in the encryption process associated with the returned user registration message.

Fig. 8 illustrates the extensions to the HTML FORM tag that are used to indicate the format of incoming (i.e., received by the web browser) forms or outgoing (i.e., transmitted by the web browser to the server) messages. The HTML FORM tag can include an outformat and an informat field. The outformat field specifies the format of the return message and the informat field specifies the format of the incoming or received form. The values for the informat field can include "encrypt" indicating that the incoming form associated with the FORM tag is encrypted. The values for the outformat field can include "encrypt", "sign", or "ensign." The "encrypt" value signifies that the form data is to be returned in an encrypted format. The "sign" value signifies that the returned form data is to include the user's digital signature and timestamp. The "ensign" value signifies that the returned form data is to be encrypted and include the user's digital signature and timestamp.

- 13 -

In addition, the key field can be used to transmit a server's encryption key along with the informat and outformat field. The server's public key is transmitted to the web browser in order for the public key to be used to encrypt the session key. In one embodiment, the server's public key can be included in each HTML form that requires encryption. In another embodiment, the server's public key can be included in the first HTML form transmitted to the user. It is retained by the web browser for the duration of the session. This obviates the need to retransmit the server's public key with each subsequent transmission.

The HTML forms emanating from the financial server can contain any combination of the above mentioned new fields in a FORM tag. Furthermore, the present invention can also accommodate a version of HTML that does not incorporate any of the new fields.

Although the present invention has been described with reference to a particular syntax for the new field in the HTML FORM tag, it is not limited to this particular embodiment. Any other syntax can be used provided that it provides a similar function. In addition, the present invention can be achieved using different HTML tags or through the use of new HTML tags.

The general architecture associated with the present invention has now been disclosed. Attention presently turns to a more detailed consideration of the architecture of the invention, the processing performed by the invention, the distinctions between the elements of the architecture, and the advantages associated with the disclosed technology.

Financial Transaction Processing Overview

Fig. 9 illustrates the steps used in the financial transaction processing system 100 and method of the present invention. Initially, a user associated with a client computer 106 installs a web browser 216 that generates a pair of encryption keys 218 (step 226). Preferably, a private/public key pair 218 is created where the private key

- 14 -

is used by the user to digitally sign forms and the public key is used by the server to authenticate the user's digital signature.

A user establishes an account 134 with the financial server 102. This account 134 is established through a user registration HTML form 264 that is transmitted to the user when the user initially accesses the financial server 102 (step 274). The user registration HTML form 264 elicits general and financial data from the user as well as the user's public key 140.

Once the user has established an account 134 with the financial server 102, the user can logon or access the account 134 at various times (step 276). During each logon session, the user and financial server 102 can exchange HTML transaction forms 124 and forms data 143 (step 278). The format of the forms varies depending on the transaction type. In some cases, the server 102 can transmit to the user an encrypted form. In other cases, the server 102 can request that the web browser 216 return a message in a number of formats. At the completion of a particular session, the user exits (step 280) and can reactivate another session with the financial server 102 at a later time.

Each of the aforementioned processes will now be described below in more detail.

The Web Browser Initialization Procedure

Initially before the financial transaction system 100 is initiated, a user installs a web browser 216 in the client computer 106. As part of the web browser's installation process, an initialization procedure 226 executes which generates one or more encryption keys 218. Preferably, a private and public key pair are generated. The private key is used to create the user's digital signature and the public key is used to verify the user's digital signature.

Referring to Figs. 10 and 11, the initialization procedure 226 displays a browser welcome page 234 (step 300). The initialization procedure 226 prompts the user for the password 232 associated with the browser 216 and verifies it (step 302). Upon

- 15 -

successful verification, the user's private/public encryption keys 218 are generated (step 304) and, optionally, displayed to the user (step 306). Any of the well known encryption techniques can be used to generate the encryption keys 218. Encryption techniques are described in Schneier, Applied Cryptography, John Wiley & Sons, 2d ed., 1996, which is hereby incorporated by reference as background information.

The initialization procedure 226 then encrypts the user's encryption keys 218 using the browser's encryption key 240 (step 308) to protect those keys from misuse or misappropriation, and stores them in a predefined location within the web browser's address space (step 310).

Financial Server Processing

Figs. 12A - 12B illustrate the steps used by the financial server 102 in processing requests and transactions from the users. A user accesses the financial server 102 by means of an account identifier 134 and a password 136 (step 320). The financial server 102 verifies the account 134 and its password 136 (step 322). This verification can be performed by matching the account 134 and password 136 against the data in the user database 132. If the financial server 102 has determined that the user is new (step 324-Y) it transmits a user registration form 264 to the user (step 326). The details of the user registration procedure is described in detail below with reference to Fig. 13.

Otherwise, the user has an established account 134 with the financial server 102 (step 324-N). The financial server 102 awaits for transmissions from the client (step 328). The transmissions can be requests for web pages, form data, as well as other types of communications. The financial server 102 identifies the type of transmission received from the client (step 330) and processes it accordingly.

If the transmission was a request for one or more web pages, the financial server 102 obtains the requested web page and transmits them to the user (step 334). The web pages can contain HTML forms that are encrypted (e.g., documents with confidential information are encrypted). The encrypted forms are identified by special fields in the

- 16 -

FORM tag as was described above previously with reference to Fig. 7. Messages with confidential user information sent by a client to the server are preferably encrypted with a session key. Forms sent by the server to a client are preferably encrypted by the server with the client's public key, because any form or document encrypted with the client's public key can be decrypted and viewed only by a person or system having access to the client's private key.

Alternately, forms containing confidential information sent by the server to a client can be encrypted with a session key generated by either the server or client. If the session key is generated by the server, the session key is encrypted with the client's public key, attached to the encrypted form being sent to the client. In this embodiment, the client first decrypts the session key with its private key and then decrypts the form with the session key. Session key encryption is often preferred for encrypting documents (as opposed to short messages) because it is typically performed using encryption techniques (such as DES) that are much less computationally expensive than public/private key encryption.

In yet another alternate embodiment, the server can utilize a different pair of public/private encryption keys for each user. In another alternate embodiment, the server can utilize a different pair of public/private encryption keys for each user for each logon session.

If the transmission is a return message, the flag field 258 of the header record 252 associated with the message is decoded in order to identify the format type (step 332). For an encrypted message (i.e., flag="E"), the server's private key 142 is used to decrypt the encrypted session key 254 (step 336). The session key was encrypted by the user's web browser 216 with the server's public key. The session key 254 is then used to decrypt the enclosed form data 256 (step 338). The server's audit trail 122 is then updated with the user's account and the form data 256 (step 340). The form data 256 is then processed accordingly (step 342).

For an encrypted message that contains the user's digital signature and timestamp

- 17 -

(i.e., flag="A"), the server's private key 142 is used to decrypt the embedded session key 254 (step 344). The session key 254 is then used to decrypt the enclosed form data 256 (step 348). Next, the user's digital signature 218 is located within the message and verified with the user's public key 140. The timestamp associated with the user's digital signature is also extracted from a predefined location within the message (step 348). The user's public key 140 is obtained from the server's user database 132. The financial server 102 searches the user database 132 using the account ID field 257 of the header record associated with the message. The audit trail 122 is then updated with the user's account, digital signature, timestamp, and the form data (step 350). The form data is then processed accordingly (step 352). The procedure then awaits for a new client communication.

An incoming message 143 that contains the digital signature of the user (i.e., flag="S") is processed using some of the same steps as a message that is encrypted with the user's digital signature and timestamp. The financial server 102 locates the digital signature of the user within the message which is verified with the user's public key 140 (step 348). The audit trail 122 is then updated with the user's account, digital signature, timestamp, and the form data (step 350). The form data is then processed accordingly (step 352). The procedure then awaits for a new client communication.

For messages that are not received in a special format (i.e., messages that are received in a non-encrypted, plain format), the audit trail 122 is updated with the user's account and the form data 256 (step 350). The form data is then processed accordingly (step 352). The procedure then awaits for a new client communication.

Server's User Registration Procedure

Fig. 13 illustrates the steps used by the web browser 216 in processing a user registration form 264 shown in Fig. 14. The web browser 216 receives an HTML document containing a user registration form 264 from the financial server 102 (step 360).

The user registration form 264 includes data entry fields that the user fills in (step

362). The web browser inserts into the registration form user information (provided by the user of the client computer and/or available to it from other resources in the client computer), typically including data that uniquely identifier the user such as the user's name, social security number or equivalent identifier, and the financial institution account number for an account previously established by the user with the financial institution associated with the server.

The web browser 216, having identified the form as the user registration form from the request field of the FORM tag (i.e., request="register"), obtains the user's public encryption key 218 and places it into a predefined location within the return message (step 364). The user's public key 218 is stored in an encrypted format in a specified location within the web browser's address space. The user's encryption keys 218 were encrypted with the browser's encryption key 240. Thus, the web browser decrypts the key from the known location with its own encryption key 240.

The web browser 216 then uses the digital signature procedures 230 to digitally sign the form data with the user's private key 218 (step 366). Next, the web browser 216 uses the random session key generation procedures 222 to generate a random session key 254 (step 368). The random session key 254 is used to encrypt the form data (step 370). The session key 254 is affixed to the top of the message and encrypted with the server's public key 142. The server's public key 142 is transmitted through the key field in the FORM tag (i.e., key = "server's public key") (step 372). The return message is then formatted using the formatting procedures 224 and transmitted to the server (step 374).

A random session key is a random-bit string generated by means of a random process. Typically, the key bits are generated from a reliably random source or a cryptographically secure pseudo-random bit generator. Any of the well known random sequence generators can be used. A description of these techniques can be found in the Schneier reference previously mentioned with respect to Figs. 10 and 11.

- 19 -

The server decrypts and verifies the information in the received registration message. If the information satisfies predefined acceptance criteria, such as matching user information associated with a previously established account at the financial institution, then a user information record is added to the user database 132. As shown in Fig. 2, the user information record identifies the user and the user's public key. Alternately (e.g., if the server's database 132 already contains user records for every user having an account at the associated financial institution), a previously existing user information record is updated to include the user's public key.

Web Browser

Once the user is registered with the financial server 102, the user, through the web browser 216, exchanges HTML transaction forms and return messages with the financial server 102. At least some of the HTML documents sent by the server to a client will contain an HTML FORM tag. In accordance with a preferred embodiment, the FORM tag includes special fields that indicate how the associated HTML document is formatted (e.g., whether or not it is encrypted). If the HTML document sent by the server to a client is encrypted, a special "key" field in the FORM tag can be used to specify the server's public key. If the form is of the type that requests user information be returned to the server, the FORM tag also includes special fields that instruct the client's Web browser as to how the return message should be formatted before it is returned to the server. The client's web browser 216 reads these FORM tag fields and performs the appropriate procedures to enable the user to read the HTML document and to properly format the message for transmission back to the server.

Figs. 15A - 15B illustrate the steps used by the web browser 216 in processing HTML documents that are received from the financial server 102. Upon receipt of an HTML document (step 380), the web browser 216 inspects the fields of the FORM tag. If the FORM tag indicates that the form associated with the tag is encrypted (i.e., the existence of an informat field) (step 382-Y), the web browser 216 recognizes that the data contained between the FORM tag pairs is encrypted. The web browser 216 reads the data from the file until it reaches the corresponding FORM tag pair (i.e.,

- 20 -

</FORM>) (step 414). The web browser 216 decrypts the form with the user's private key 218 (step 416) and continues to read the tags in the HTML document.

Next, the web browser 216 detects if the FORM tag has an outformat field (step 384). If the FORM tag does not include an outformat field (step 384-N), the form is displayed and processed accordingly (step 386). Otherwise (step 384-Y), the outformat field is stored and the form is displayed and processed accordingly (step 386).

Once the form has been processed, the web browser 216 prepares the return message in accordance with the requested directives specified in the stored outformat field (step 390).

If the requested return message format specified encryption with a digital signature and timestamp (i.e., outformat = "ensign"), the web browser 216 digitally signs the form data with the user's private key 218 and appends a timestamp at a predefined location within the message (step 392). Next, the web browser 216 randomly generates a session key 254 (step 394). The message is then encrypted with the randomly generated session key 254 (step 396). In some embodiments, a single session key is used for all encrypted client message transmissions during a single session, in which case step 396 is skipped after the transmission of the first encrypted client message during a session.

The session key 254 is encrypted with the server's public key 142 and affixed to the encrypted message (step 398). As noted above with respect to Fig. 8, the server's encryption key is transmitted to the web browser either initially with the first transmitted HTML form or with each transmitted HTML form requiring encryption.

A header record 252 is then generated containing a flag 258 having the appropriate value ("A") and the key length 260 of the enclosed session key 254. The message is formatted and then transmitted to the financial server 102 (step 400).

- 21 -

If the requested return message format specified encryption (i.e., outformat = "encrypt"), the web browser 216 performs some of the same steps described above. The web browser 216 randomly generates a session key 254 (step 394). The form data is then encrypted with the randomly generated session key 254 (step 396). The session key 254 is affixed to the encrypted form data and encrypted with the server's public key 142 (step 398). A header record 252 is then generated containing a flag 258 having the appropriate value ("E") and the key length 260 of the enclosed session key. The message is formatted and then transmitted to the financial server 102 (step 400).

If the requested return message format specified the user's digital signature (i.e., outformat = "sign"), the web browser 216 signs the form data with the user's private key 218 (step 410). In addition, a timestamp is generated and appended to the form data (step 410). A header record 252 is then generated containing the appropriate flag value ("S"). Since no session key is enclosed in the message, the key length field is blank. The message is formatted and then transmitted to the financial server 102 (step 412).

In conclusion, the aforementioned description describes a method and system for securely transmitting transactions embodied as HTML forms between a financial server and a web browser. The technology of the present invention incorporates five security features to ensure that only the intended parties of the transaction securely receive and transmit a transaction. The five security features include: privacy, in the form of session key encryption; data integrity, through the use of data encryption; access control, via a password mechanism; user nonrepudiation, by means of digital signatures and timestamps; and a server side audit trail. These security features are embedded in the financial server and web browser in an automatic and transparent manner.

- 22 -

Alternate Embodiments

While the present invention has been described with reference to a few specific embodiments, the description is illustrative of the invention and is not to be construed as to limiting the invention. Various modifications may occur to those skilled in the art without departing from the true spirit and scope of the invention as defined by the appended claims.

Further, the method and system described hereinabove is amenable for execution on various types of executable mediums other than a memory device such as a random access memory. Other types of executable mediums can be used, such as but not limited to, a computer readable storage medium which can be any memory device, compact disc, or floppy disk.

Although the present invention has been described with reference to encryption and digital signature techniques that utilize encryption key pairs, it is not limited to this particular technique. Any technology or technique that provides similar functionality can be utilized.

Furthermore, one skilled in the art can easily modify the present invention to incorporate a digital signature mechanism in the HTML forms that are transmitted to a web browser. Moreover, additional security features can be easily added to either the client or server side of the transaction processing.

- 23 -

WHAT IS CLAIMED:

1. A computer-implemented method for transmitting financial transactions between at least one client computer and at least one server computer interconnected by a communications link, the method comprising the steps of:
 - (a) receiving one or more HTML documents from the server computer, a subset of the documents including format directives, each of a first subset of the format directives indicating an outgoing transmission format used in transmitting a return message to the server computer, each of a second subset of the format directives indicating an incoming transmission format used to receive an HTML document, each of the first and second subsets of format directives associated with at least one cryptographic technique;
 - (b) inserting form data from a received HTML document into said return message;
 - (c) applying one or more cryptographic techniques to the form data, each applied cryptographic technique identified in a respective format directive associated with the received HTML document; and
 - (d) transmitting the return message to the server computer.
2. The method of claim 1,
the format directives selected from the set consisting of encryption, digital signature and timestamp, and encryption with digital signature and timestamp.
3. The method of claim 1,
step (c) further including the steps of:
 - (1) generating a first encryption key;
 - (2) encrypting the form data, including any inserted user related information, with the first encryption key;
 - (3) affixing the first encryption key to the return message; and
 - (4) encrypting the first encryption key with a second encryption key.
4. The method of claim 3,

- 24 -

step (c)(1) further including the step of utilizing a random key sequence generator to generate the first encryption key.

5. The method of claim 3,
step (c)(4) further including the step of obtaining the second encryption key from the server computer.
6. The method of claim 3,
step (c)(1) further including the step of:
prior to the generating step, storing a digital signature associated with a specified user in the return message.
7. The method of claim 6,
step (c) further including the steps of:
generating a digital signature and a digital signature verifier; and
providing the server computer with the digital signature verifier.
8. The method of claim 1,
step (c) further including the step of storing a timestamp in the return message.
9. A web browser for accessing data within a computer system including at least one client computer connected through a communications link with at least one server computer, the browser comprising:
a memory for storing a plurality of HTML documents, a subset of the HTML documents including cryptographic protocol directives for use in returning form data included in said HTML documents;
browsing mechanism for retrieving various ones of the HTML documents from the server and for inserting user related information in said form data which is included in a return message; and
a cryptographic processing mechanism for processing the return message and HTML documents in accordance with a specified cryptographic protocol;
wherein the web browser utilizes the browsing mechanism to receive and

- 25 -

transmit the return message to and from the server computer and utilizes the cryptographic processing mechanism to exercise one or more cryptographic protocols on the HTML documents and return message in order to enable receipt of the HTML documents by an intended user associated with the client computer and to provide secure transmission of the return message transmitted to the server computer.

10. The web browser of claim 9,
the cryptographic processing mechanism including an encryption processing mechanism for encrypting the return message and decrypting HTML documents.
11. The web browser of claim 9,
the cryptographic processing mechanism including a digital signature processing mechanism for signing the return message and authenticating HTML documents.
12. The web browser of claim 11,
the digital signature processing mechanism including a timestamp processing mechanism.
13. The web browser of claim 9,
the cryptographic processing mechanism including an encryption key generation mechanism for creating one or more encryption keys.
14. The web browser of claim 13,
the encryption key generation mechanism including a random key generation mechanism for creating one or more random session keys.
15. A computer readable storage medium containing a computer code mechanism for use with a financial server, the computer code mechanism comprising:
a browsing mechanism for retrieving HTML documents from the financial server, said HTML documents including form data, said browsing mechanism inserting user related information in said form data which is appended to a return

- 26 -

message that is transmitted to the financial server, a subset of the HTML documents retrieved from the financial server including cryptographic protocol directives for use in exchanging return messages to the financial server; and

a cryptographic processing mechanism for processing the HTML documents in accordance with a specified cryptographic protocol;

wherein the computer code mechanism utilizes the browsing mechanism to receive HTML documents from the financial server and to transmit return messages to the financial server and utilizes the cryptographic processing mechanism for exercising one or more cryptographic protocols on the HTML documents in order to enable receipt of the received HTML documents by an intended user and to ensure secure transmission of the return message transmitted to the financial server.

16. The medium of claim 15,

the cryptographic processing mechanism including an encryption processing mechanism for decrypting HTML documents and encrypting the return message.

17. The medium of claim 15,

the cryptographic processing mechanism including a digital signature processing mechanism for signing the return message and authenticating the HTML documents.

18. The medium of claim 17,

the digital signature processing mechanism including a timestamp processing mechanism.

19. The medium of claim 15,

the cryptographic processing mechanism including an encryption key generation mechanism for creating one or more encryption keys.

20. The medium of claim 19,

the encryption key generation mechanism including a random key generation mechanism for creating one or more random session keys.

- 27 -

21. A computer network for financial transaction processing, the network comprising:

- a plurality of client computers, each client computer associated with one or more users;

- at least one financial server comprising:

- a memory for storing a plurality of HTML documents representing financial transactions, each HTML document including form data, a subset of the HTML documents including cryptographic protocol directives for use in exchanging financial transactions between the client computers and the server computer;

- one or more cryptographic processing mechanisms for use in encoding said form data and decoding each received HTML document; and

- a server mechanism for managing communications from the client computers, a subset of the communications including a return message including the form data, the server mechanism including instructions to interpret the cryptographic protocol directives associated with each received return message and to process each received return message in accordance with one or more corresponding cryptographic protocol mechanisms.

22. The network of claim 21,

- the cryptographic processing mechanism including:

- an encryption processing mechanism for encrypting the form data and decrypting the HTML documents;

- a user information database that includes user public key information associated with each user registered to exchange confidential information with the financial server;

- a digital signature verification mechanism for verifying a digital signature in each digitally signed communication received from a client computer in accordance with the corresponding digital signature, if any, stored in the user information database; and

- an audit trail for storing records of digitally signed communications received from client computers sufficient to prove that such each received communication was digitally signed by a respective user registered to exchange

- 28 -

confidential information with the financial server.

23. The network of claim 22,
the cryptographic protocol directives selected from the set consisting of encryption, digital signature and timestamp, and encryption with digital signature and timestamp.
24. The network of claim 22,
each client computer including a web browser for accessing HTML documents from the financial server.
25. The network of claim 24,
the web browser including a cryptographic processing mechanism for encrypting form data and decrypting the accessed HTML documents.
26. The network of claim 24,
the web browser including a digital signature processing mechanism for signing the return message and authenticating HTML documents.
27. The network of claim 26,
the digital signature processing mechanism further including a timestamp processing mechanism.
28. The network of claim 24,
the web browser including an encryption key generation mechanism for creating one or more encryption keys.

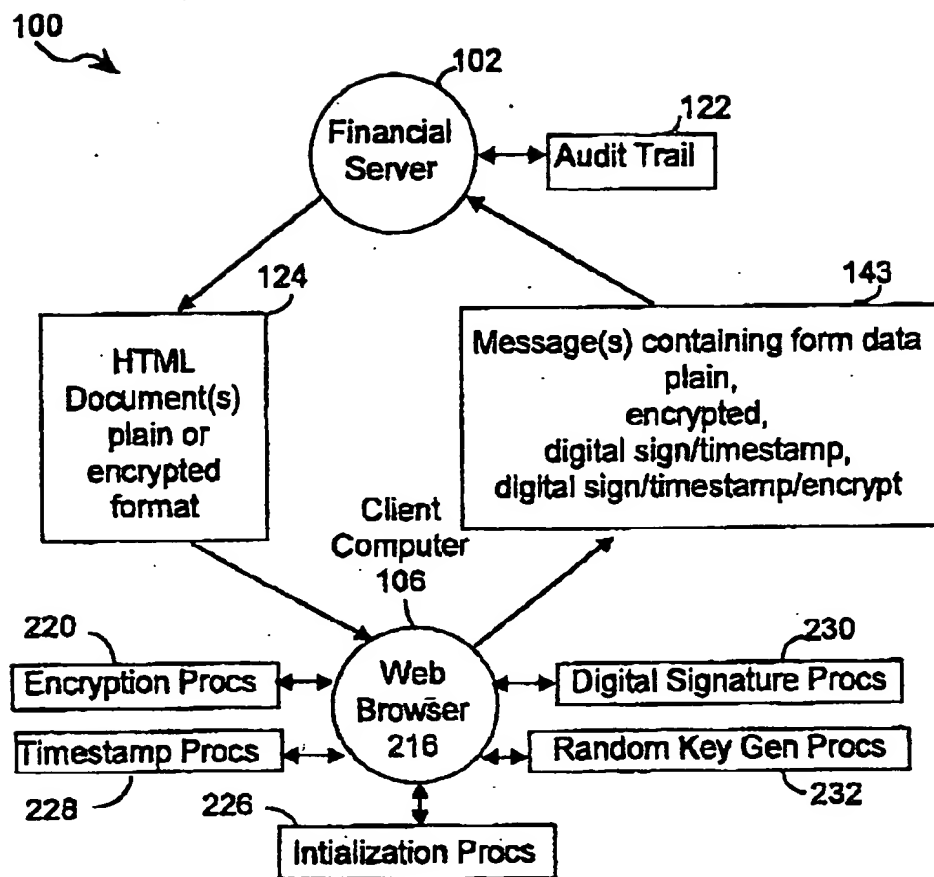


FIG. 1

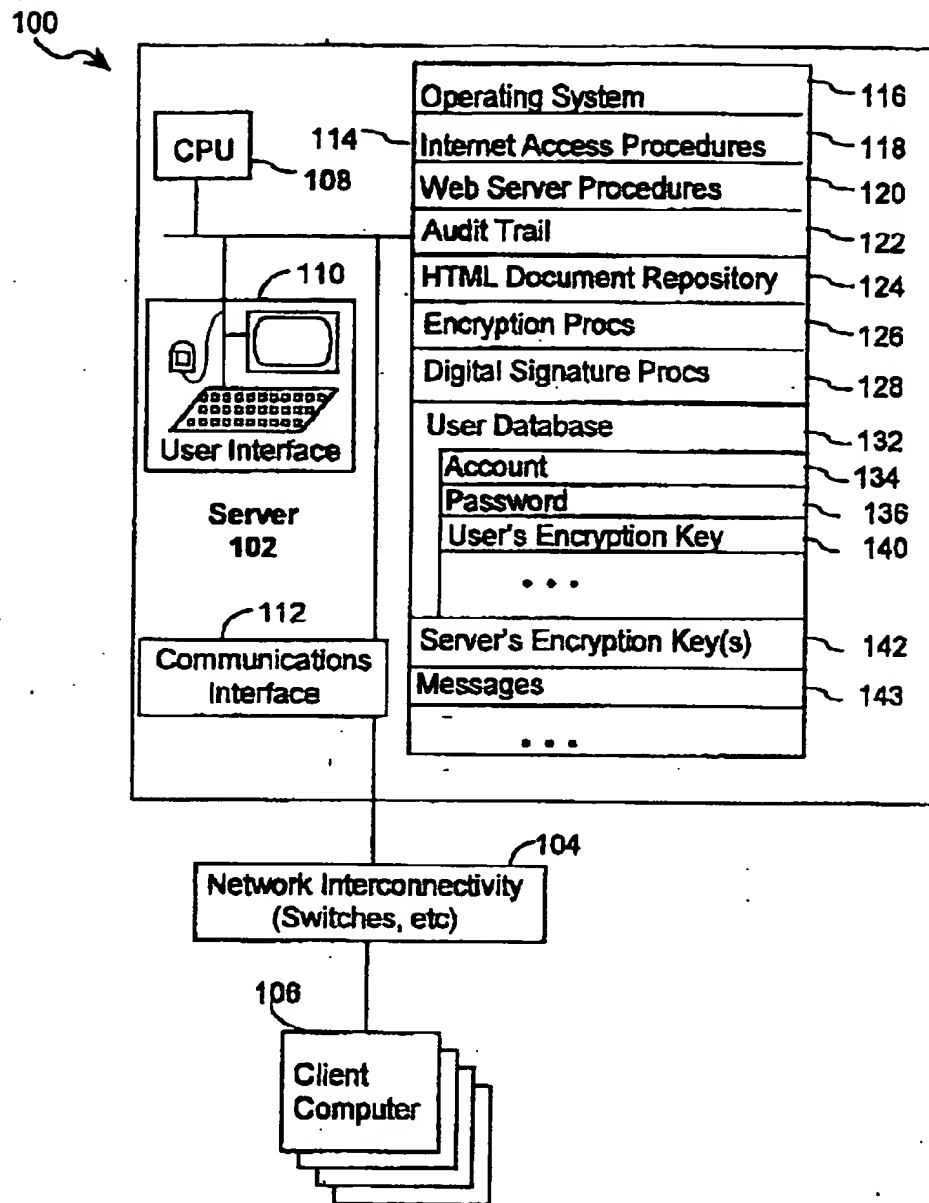


FIG. 2

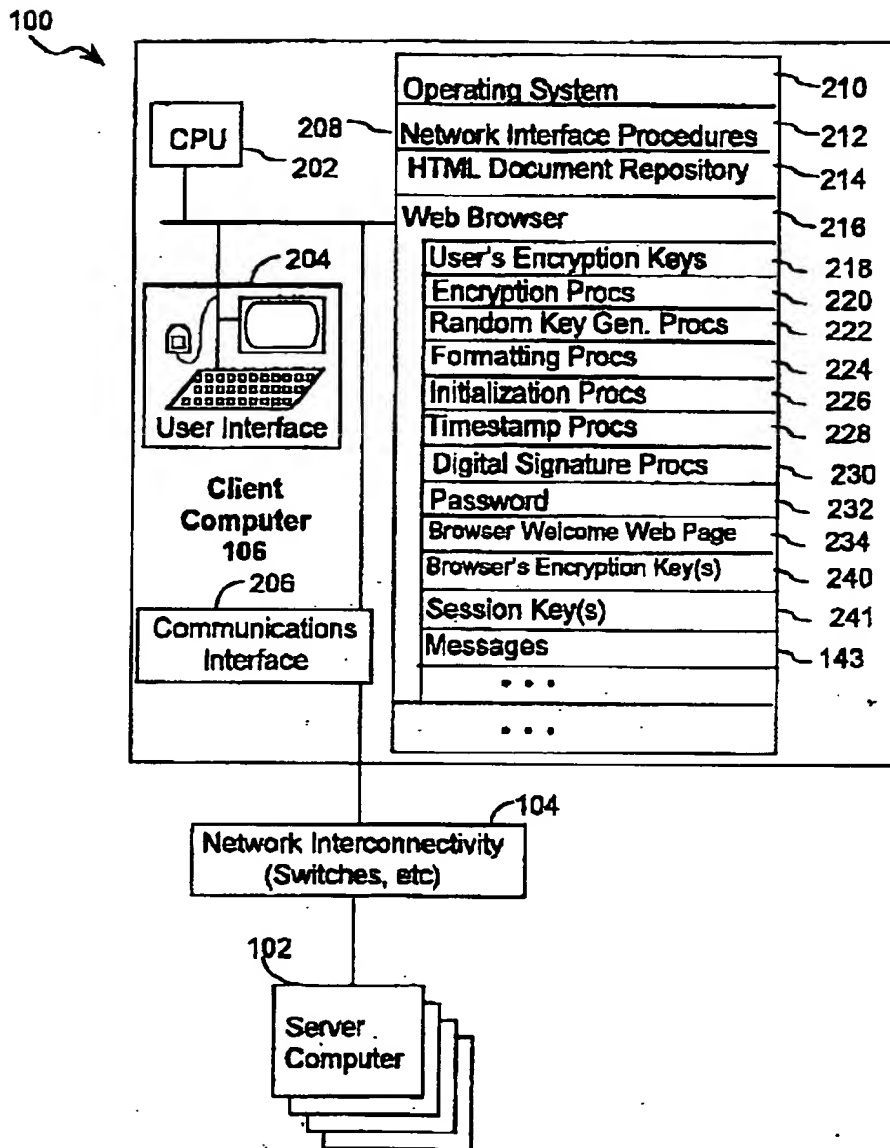


FIG. 3

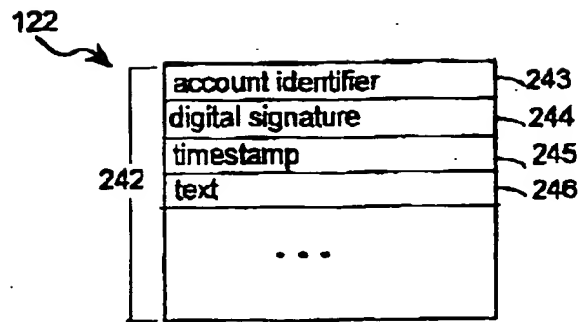


FIG. 4

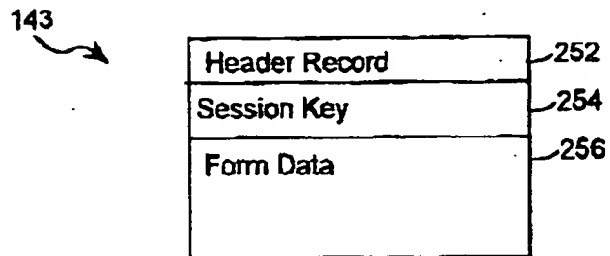


FIG. 5

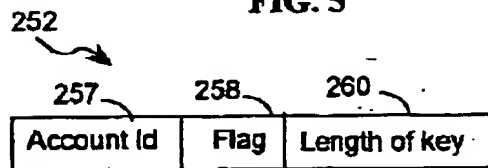
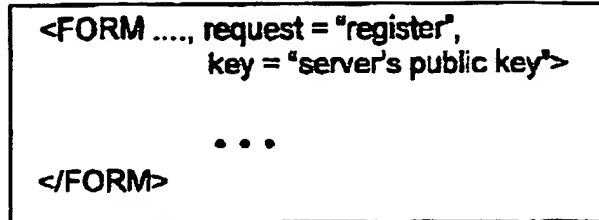


FIG. 6

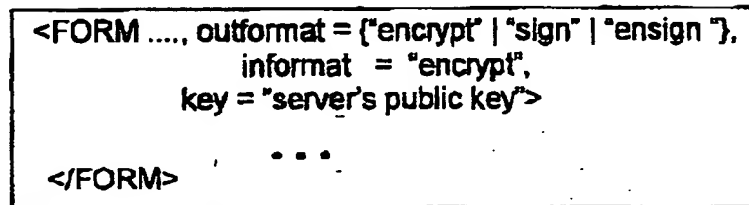
268



```
<FORM ....., request = "register",  
                key = "server's public key">  
    ...  
</FORM>
```

FIG. 7

270



```
<FORM ....., outformat = {"encrypt" | "sign" | "ensign"},  
                informat = "encrypt",  
                key = "server's public key">  
    ...  
</FORM>
```

FIG. 8

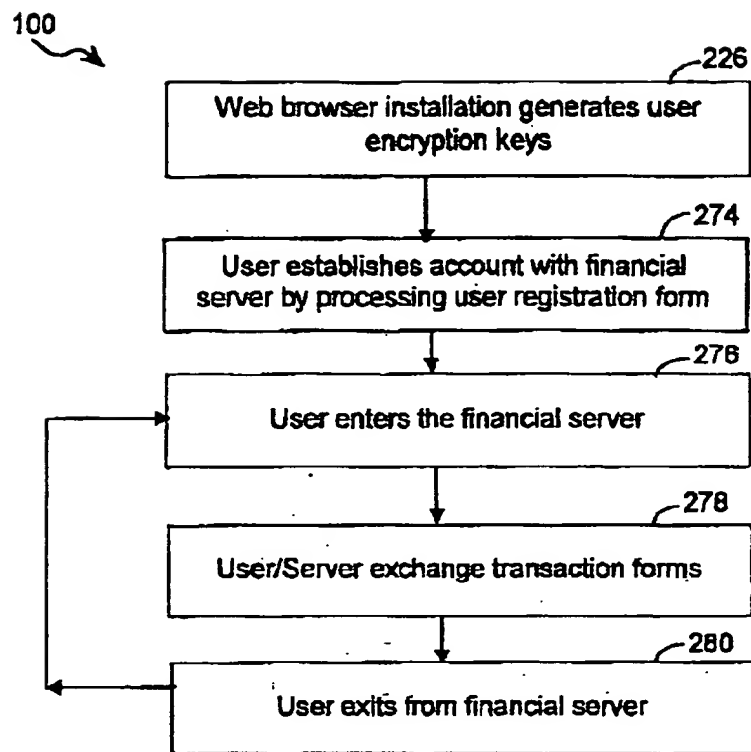


FIG. 9

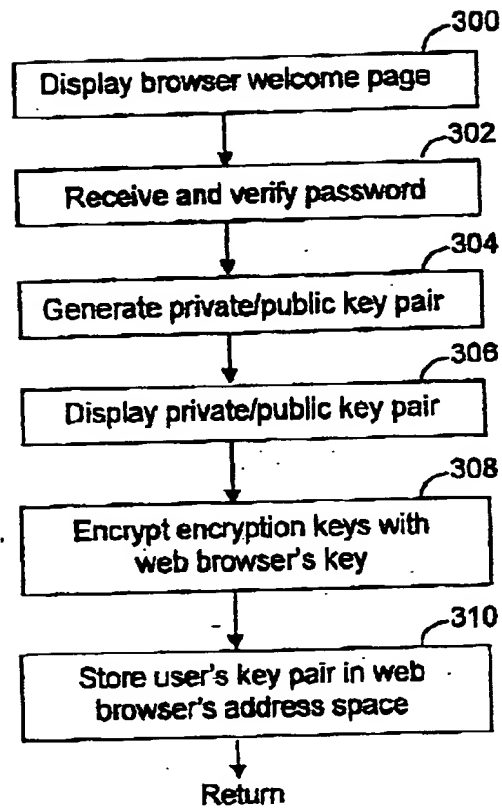
226
→

FIG. 10

234
→

BROWSER WELCOME PAGE

Establish Encryption key pair

Enter password:

Encryption Keys:

<input type="text"/>	private key
<input type="text"/>	public key

FIG. 11264
→

USER REGISTRATION FORM

Name:

Address:

...

FIG. 14

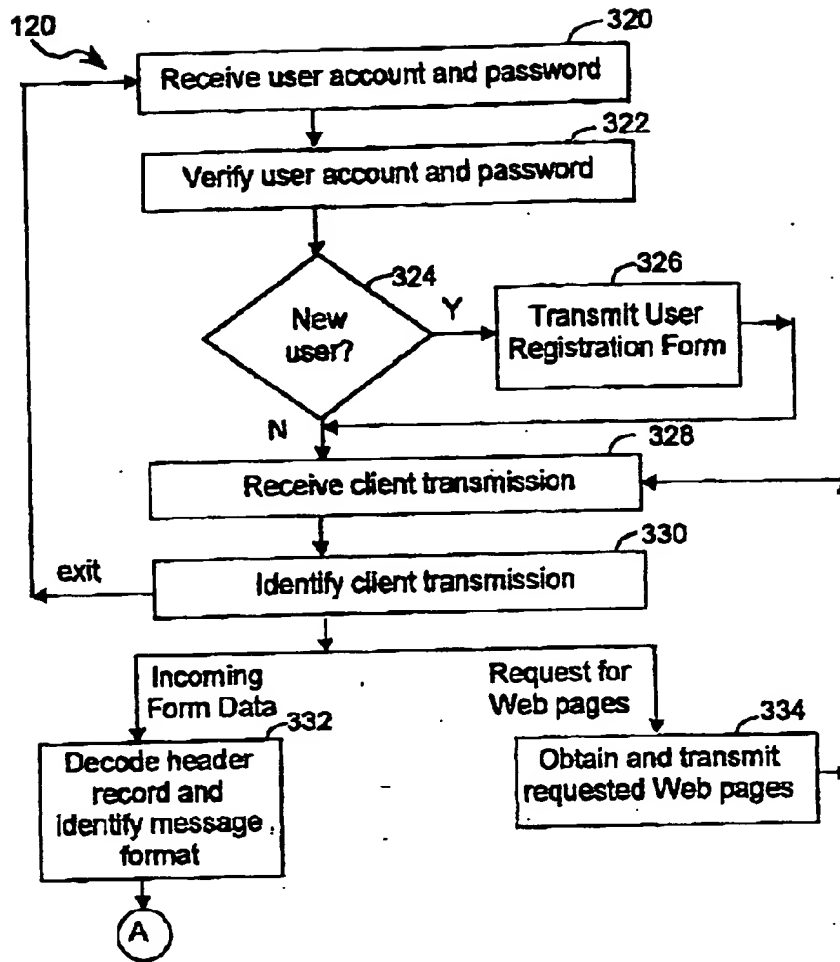


FIG. 12A

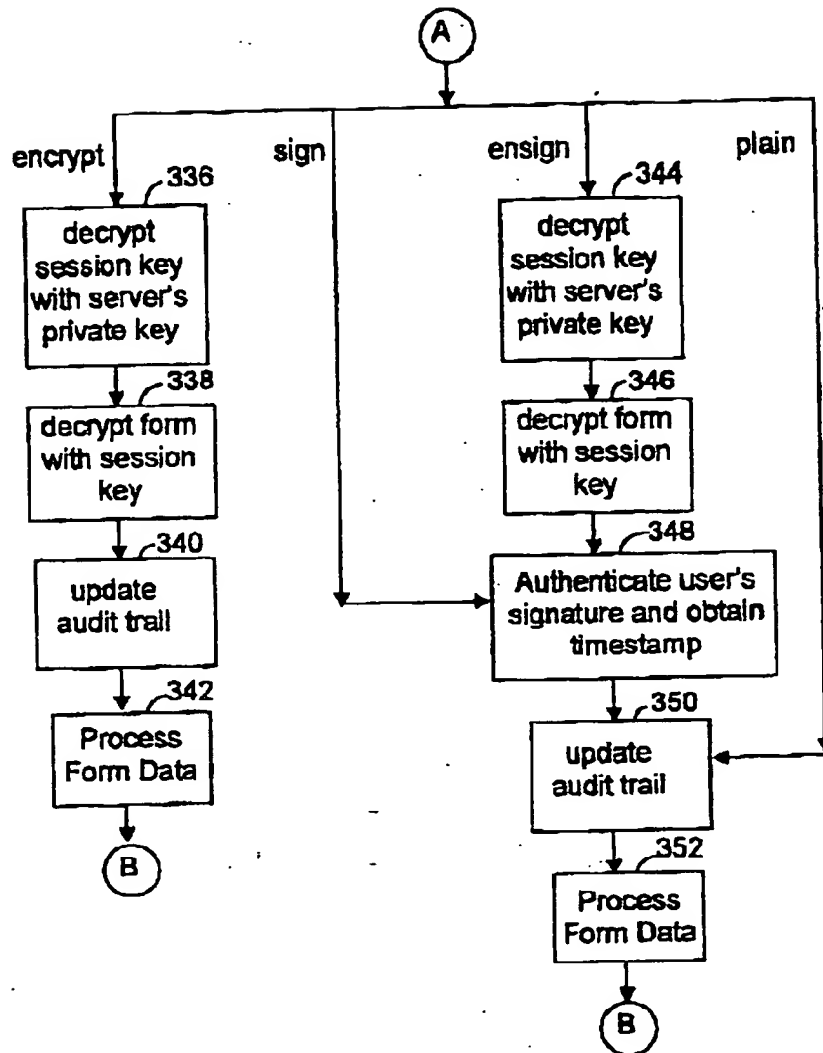


FIG. 12B

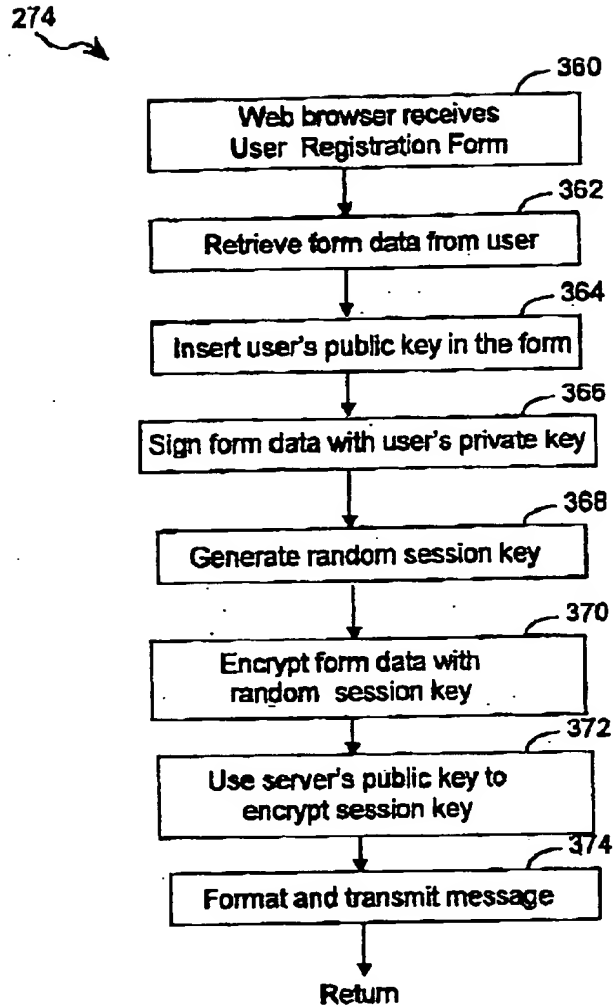


FIG. 13

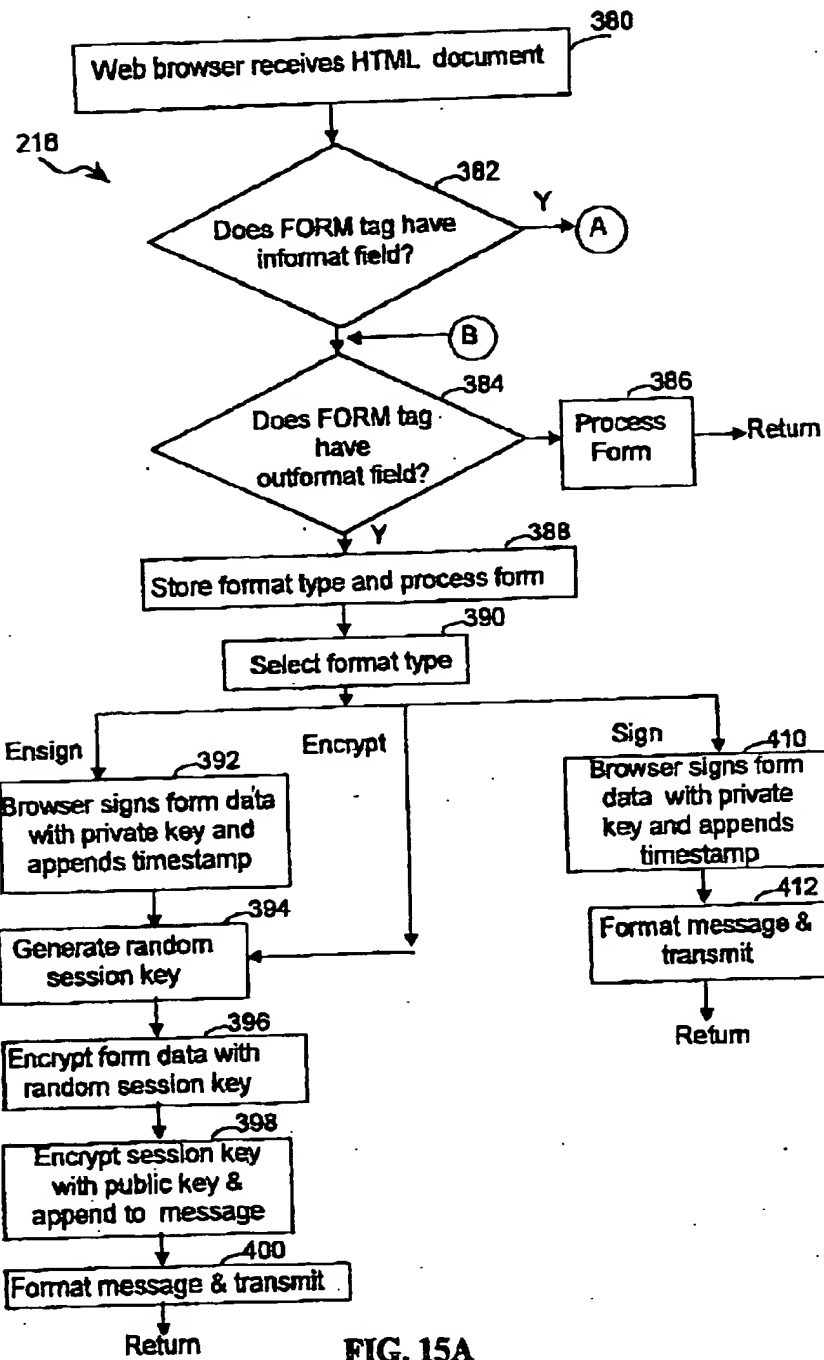
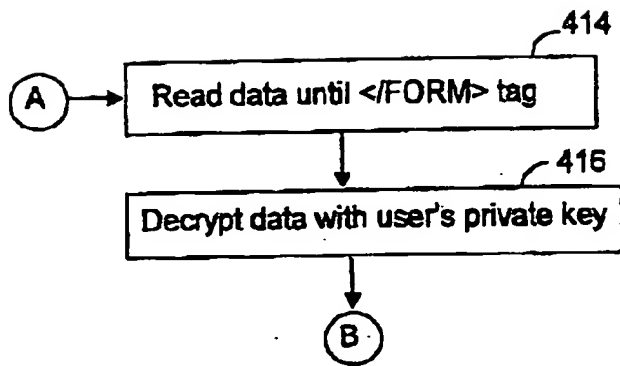


FIG. 15A

**FIG. 15B**

ABSTRACT OF THE DISCLOSURE

The financial transaction processing system includes at least one financial server connected through a public network to a number of users associated with client computers. Each user accesses the financial server through a web browser. The web browser is provided with the capabilities to generate encryption keys, to encrypt and decrypt HTML forms, and to digitally sign and timestamp HTML forms. The financial server transfers web pages including HTML forms representing financial transactions. The HTML forms contain extensions that specify the format of an incoming format and the format of a returned form. An HTML form can be transmitted in an encrypted format, in a format including a user's digital signature and timestamp, and in an encrypted format that contains the user's digital signature and timestamp. The financial server tracks each processed transaction through an audit trail including the user's account, the user's digital signature, the timestamp of the transaction, and the text of the transaction.

**This Page is Inserted by IFW Indexing and Scanning
Operations and is not part of the Official Record**

BEST AVAILABLE IMAGES

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images include but are not limited to the items checked:

- ☐ BLACK BORDERS
- ☐ IMAGE CUT OFF AT TOP, BOTTOM OR SIDES
- ☐ FADED TEXT OR DRAWING
- ☐ BLURRED OR ILLEGIBLE TEXT OR DRAWING
- ☐ SKEWED/SLANTED IMAGES
- ☐ COLOR OR BLACK AND WHITE PHOTOGRAPHS
- ☐ GRAY SCALE DOCUMENTS
- ☐ LINES OR MARKS ON ORIGINAL DOCUMENT
- ☒ REFERENCE(S) OR EXHIBIT(S) SUBMITTED ARE POOR QUALITY
- ☐ OTHER: _____

IMAGES ARE BEST AVAILABLE COPY.

As rescanning these documents will not correct the image problems checked, please do not report these problems to the IFW Image Problem Mailbox.